

Komponentenorientierte Modellierung elektrischer Systeme in der Mechatronik

Dissertation

zur Erlangung des akademischen Grades

Doktoringenieur (Dr.-Ing.)

von M. Sc. Yevgen Yefimenko

geb. am 23.09.1977 in Berdsk (Russland)

genehmigt durch die Fakultät für Maschinenbau

der Otto-von-Guericke-Universität Magdeburg

Gutachter:

Prof. Dr.-Ing. Roland Kasper

Doz. Dr. Alexander Derkach

Promotionskolloquium am 14.10.2005

Vorwort

Die vorliegende Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Institut für Mechatronik und Antriebstechnik (IMAT) der Otto-von-Guericke-Universität Magdeburg.

Mein besonderer Dank gilt dem Leiter des Instituts, Herrn Prof. Dr.-Ing. R. Kasper für die Betreuung und Unterstützung der vorliegenden Dissertation. Er gab mir Gelegenheit, diese Arbeit am Lehrstuhl für Mechatronik durchzuführen und hat sie stets durch anregende Diskussionen und wertvolle Ratschläge begleitet.

Ich danke dem Gutachter der vorliegenden Dissertation, Herrn Dr.-Ing. A. Derkach für sein Engagement und seinen Einsatz.

Für ihre Anregungen und Motivationen möchte ich allen Mitarbeitern des Instituts, die mich wo immer möglich unterstützt haben, danken. Frau Dipl.-Ing. (FH) Manuela Rudolph und Herrn Dr.-Ing. Hans-Georg Baldauf danke ich für ihre Hinweise bei der Ausführung der schriftlichen Arbeit. Insbesondere möchte ich mich bei meinem ehemaligen Kollegen Herrn Dipl.-Ing. Andreas Wolf bedanken.

Mein Dank gilt meiner Familie, Prof. Dr.-Ing. habil. V. Kalashnikov, Dr.-Ing. A. Chorchordin und meinen Freunden, ohne deren Unterstützung und Verständnis diese Arbeit nicht möglich gewesen wäre.

Magdeburg, im August 2005

Yevgen Yefimenko

Inhaltsverzeichnis

Abkürzungen	V
Mathematische Symbolik	V
Formelzeichen und Symbole	V
Kurzfassung	VI
1 Einleitung	1
1.1 Einleitung	1
1.1.1 Grundbegriffe der Modellierung und Simulation	1
1.1.2 Grundbegriffe der Mechatronik	2
1.1.3 Hintergründe und Motivation	3
1.2 Stand der Wissenschaft und Technik	5
1.2.1 Modellierung elektrischer Systeme	5
1.2.2 Modellierung elektro-mechanisch gekoppelter Systeme	6
1.2.3 Modellierungssprachen und symbolverarbeitende Tools	6
1.2.4 Bondgraphen	7
1.2.5 Blockdiagramme	9
1.2.7 Komponentenorientierte Modellierung	10
1.3 Ziele und Inhalt der Arbeit	11
2 Theoretische Grundlagen der Methode der komponentenorientierten Modellierung elektrischer Systeme	14
2.1 Idee der Methode	14
2.1.1 Darstellung des Systems	14
2.1.2 Definition funktionaler Strukturteile	15
2.1.3 Einteilung hierarchischer Ebenen	15
2.1.4 Bildung der Komponente	16
2.1.5 Simulation	17
2.2 Definition der Elektrischen Schnittstelle	17
2.2.1 Allgemeine Annahmen	17
2.2.2 Definition der zweipoligen Widerstandskomponente	18

2.2.3	Definition der zweipoligen reaktiven Komponente	26
2.2.4	Strukturkomponente mit mehreren Eingängen	29
2.3	Modellierung der Grundkomponenten	31
2.3.1	Modelle der Widerstandskomponenten.....	31
2.3.2	Modelle der Komponente mit den Zuständen	34
2.3.3	Modell des MOSFET Transistors und des Schalters	37
2.3.5	Modell des Piezoaktors	40
2.3.6	Modellierung der Diode	42
2.4	Allgemeine Methodik der Berechnung der Schnittstellenparameter	45
2.4.1	Die Inzidenzmatrix eines gerichteten Graphen und Stromgleichungen.....	46
2.4.2	Formulierung der Spannungsgleichungen.....	49
2.4.3	Sparse-Tableau Formulierung der Netzwerkgleichungen.....	51
2.4.4	Lösung der Tableau-Matrizengleichung	53
2.4.5	Ermittlung der Schnittstellenparameter.....	55
2.4.6	Anwendung der Parameterberechnungsmethodik bei der Strukturkomponente.....	61
2.4.7	Kurze Zusammenfassung der Methodik der Parameterermittlung	62
2.5	Berechnung der Unbekannten	64
2.5.1	Reihenfolge der Berechnung.....	64
2.5.2	Berechnung der Schnittstellenströme auf der obersten Ebene	70
2.5.3	Berechnung der Unbekannten in der Komponente	78
3	Spezielle Punkte der Modellierung und Ablauf der Modellierung	83
3.1	Behandlung der Schaltvorgänge.....	83
3.2	Änderung der Reihenfolge der Berechnung der Unbekannten	87
3.3	Ablauf der Modellierung.....	88
3.4	Modellierung des Netzwerkes mit einem Schalter.....	91
3.4.1	Aufgabenstellung	91
3.4.2	Bildung des komponentenorientierten Modells des Netzwerkes	92
3.4.3	Simulation	95
4	Eingabe des Netzwerkes.....	99
4.1	Struktur der SPICE – Netzliste.....	99
4.2	Annahmen und Erweiterung der PSPICE – Netzliste für die Zwecke der komponentenorientierten Methode	103

5 Modellierung des Schaltverstärkers der Piezoansteuerung.....	105
5.1 Schaltverstärker in Form einer Ladungspumpe	105
5.2 Bildung der Komponenten der Ladungspumpe	107
5.2.1 Bildung der Komponente <i>UP</i>	107
5.2.2 Bildung der Komponente <i>DN</i>	108
5.2.3 Bildung der Komponente <i>DRIVER</i>	110
5.2.4 Simulation	111
6 Zusammenfassung.....	116
6.1 Ergebnisse	116
6.2 Ausblick	118
Literatur.....	120
Anhang A Symbolische Lösung der Gleichungen des Netzwerks mit einem Schalter	125
Anhang B: Elementdeklaration in SPICE	129

Abkürzungen

IMAT	Institut für Mechatronik und Antriebstechnik an der Universität Magdeburg
VHDL	Very High Speed Integrated Circuit Hardware Description Language
VSD	“Visual System Developer” – interner Arbeitsname der aktuellen Implementierung der interaktiven Entwurfsplattform des IMAT

Mathematische Symbolik

s, S	Skalare Größe (Kursivschrift)
\mathbf{M}	Matrix oder Vektor (Großbuchstabe, fett)
$Y(s)$	Größe im Bildbereich
R	Größe im Zeitbereich
v	Beliebiges Element des Vektors, der Matrix oder der Menge (Index)
f	Beliebige Funktion

Formelzeichen und Symbole

Alle Formelzeichen und Symbole werden bei ihrer erstmaligen Verwendung im laufenden Text erklärt, ebenso die verwendeten Indizes. Die wichtigsten Zeichen werden in der folgenden Aufstellung ohne Anspruch auf Vollständigkeit zusammengefasst. Wenn möglich wird auch die Vorzugseinheit in der entwickelten technischen Software angegeben.

C	elektrische Kapazität [F]
I	elektrischer Strom [A]
U, V	elektrische Spannung [V]
L	elektrische Induktivität [H]
P	elektrisches Potential [V]
Q	elektrische Ladung [C]
R	elektrischer Widerstand [Ω]

Kurzfassung

Das Zusammenwachsen von Mechanik, Elektronik und Informatik unter dem Dach der Mechatronik hat in den letzten Jahren zu einer großen Zahl wissenschaftlich sehr interessanter und wirtschaftlich sehr erfolgreicher Produkte geführt. Die Integration neuer Funktionen in ein Produkt über ursprünglich bestehende Fachgebietsgrenzen hinweg legt dabei völlig neue Freiheitsgrade offen, welche im Rahmen der Entwicklung mechatronischer Produkte genutzt werden können, aber auch genutzt werden müssen, um die geforderten Produkteigenschaften darzustellen. Eine durchgängige, alle Domänen umfassende entwicklungsbegleitende Simulation wird allgemein als eine der wichtigsten methodischen Grundlagen moderner Entwicklungsprozesse gesehen. Eine sehr breite Methoden- und Werkzeuglandschaft zur Modellierung und Simulation mechanischer und elektrischer – teilweise auch elektromechanischer – Systeme ist vorhanden und wird auch breit genutzt.

Keines der Werkzeuge und vor allem nicht die darunter liegenden Methoden kann die Basis für eine notwendige durchgängige und domänenübergreifende Unterstützung des Entwicklungsprozesses mechatronischer Systeme und Komponenten bilden.

Ziel und Inhalt dieser Arbeit besteht im Aufzeigen der Besonderheiten und in der Herleitung von Lösungsansätzen zur Modellierung elektrischer Teilsysteme im Kontext eines bis zur Code-Ebene konsequent komponentenorientierten computerbasierten Werkzeugs zum Entwurf und zur Simulation mechatronischer Systeme.

Bei der komponentenorientierten Modellierung wird das Gesamtsystem als Menge miteinander verbundener Objekte dargestellt. Mechatronische Objekte wirken mit Hilfe spezieller Schnittstellen zusammen, die realen physikalischen Schnittstellen der Produkte nachgebildet sind. Aus informationstechnischer Sicht wird das modellierte System auf hierarchische Ebenen aufgeteilt. Die Parameter und notwendige algebraische und Differenzialgleichungen werden für die Komponenten generiert. Das System der Gleichungen wird symbolisch vereinfacht und gelöst, um die Dimension der Matrix und Anzahl der Unbekannten zu reduzieren. Nach der Generierung und dem Einbringen in eine Bibliothek steht die neue Komponente für die Simulation im Rahmen des Systementwurfs eines mechatronischen Systems zur Verfügung. Die Entwicklung der Schnittstellen und deren Wechselbeziehung mit dem Modellverhalten wird detailliert erklärt. Es wird versucht die Stärken und Einsatzmöglichkeiten sowie die Grenzen der entwickelten Lösung aufzuzeigen.

1 Einleitung

1.1 Einleitung

1.1.1 Grundbegriffe der Modellierung und Simulation

Modellierung und Simulation haben mit der Einführung der Informationstechnik in den letzten fünfzig Jahren einen Großteil der Ingenieur- wie auch der Natur-, und Sozialwissenschaften durchdrungen. Die Modelle werden an Stelle der realen Systeme herangezogen, um die inneren Zusammenhänge eines Systems zu begreifen und Prognosen über dessen Verhalten zu machen. Dabei wird die Simulation häufig auch als Ersatz für Experimente an einem existierenden System eingesetzt, was eine ganze Reihe von Vorteilen mit sich bringt [PEL01]:

- Im Vergleich zu realen Experimenten erfordern virtuelle Experimente häufig einen erheblich geringeren finanziellen und zeitlichen Aufwand, da die Modellierung virtueller Prototypen in der Regel deutlich weniger aufwändig ist als der Aufbau realer Prototypen.
- Mancher Systemzustand lässt sich im realen System nicht oder nicht zerstörungsfrei herbeiführen.
- Im Normalfall sind virtuelle Experimente in allen Aspekten wiederholbar, was sich für das reale System nicht, oder nur mit erheblichem Aufwand sicherstellen lässt.
- Simulierte Modelle sind in der Regel vollständig kontrollierbar. Damit kann man alle Eingangsgrößen und Parameter des Systems entsprechend vorgeben. Für ein reales System gilt dies üblicherweise nicht.
- Simulierte Modelle sind in der Regel vollständig beobachtbar. Alle Ausgangsgrößen und internen Zustände stehen zur Verfügung, während im realen System jede zu beobachtende Größe zumindest einen erheblichen Messaufwand nach sich zieht. Zudem beeinflusst jede Messung das Systemverhalten.

Diesen Vorteilen stehen einige Nachteile gegenüber:

- Vor jedem virtuellen Experiment ist eine vollständige, validierte und verifizierte Modellierung des Systems vorzunehmen.
- Die Detailtreue und damit die Simulationsgeschwindigkeit der eingesetzten Modelle sind durch die Leistungsfähigkeit der für die Simulation eingesetzten Rechner begrenzt.

In vielen Fällen überwiegen die Vorteile, so dass sich virtuelle Experimente mit Gewinn einsetzen lassen. Die durch den Computer sichergestellte Wiederholbarkeit ist besonders vorteilhaft, wenn die virtuellen Experimente im Rahmen einer Optimierung systematisch geplant und durchgeführt werden.

1.1.2 Grundbegriffe der Mechatronik

Der Begriff „*Mechatronik*“ (engl. Mechatronics) setzt sich aus den beiden Bestandteilen **Mechanik** und **Elektronik** zusammen. Er wurde 1969 durch die japanische Firma Yaskawa Electric Cooperation geprägt und ab 1971 von dieser Firma als Handelsname geschützt. Ursprünglich war damit die Ergänzung mechanischer Komponenten durch Elektronik in der Gerätetechnik gemeint. Ein typisches Beispiel hierfür ist die Entwicklung moderner Spiegelreflexkameras. Seit 1982 ist der Begriff frei verfügbar. Heute ist mit diesem Wort eine Ingenieurwissenschaft verbunden, die auf den klassischen Disziplinen Maschinenbau, Elektrotechnik und Informatik aufbaut und deren Ziel die Verbesserung der Funktionalität eines technischen Systems durch ihre Verknüpfung ist (Bild 1.1) [HEI98].

Aus diesen Darlegungen wird klar, dass die Mechatronik interdisziplinären Charakter besitzt und die folgenden Gebiete umfasst:

- Mechanik (Technische Mechanik, Maschinenbau, Feinwerktechnik)
- Elektrotechnik/Elektronik (Mikroelektronik, Leistungselektronik, Messtechnik, Aktorik),
- Informationsverarbeitung (Systemtheorie, Prozessdatenverarbeitung, künstliche Intelligenz).

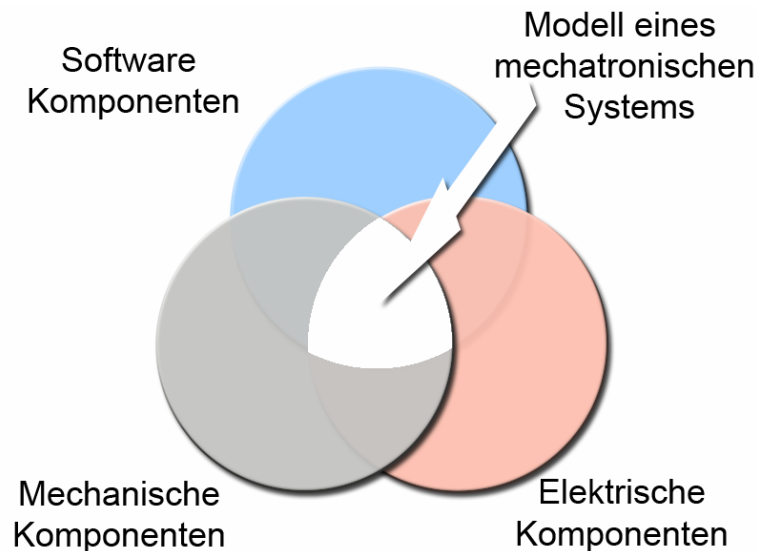


Bild 1.1: Bestandteile des Modells eines mechatronischen Systems

Wichtige physikalische Größen in mechatronischen Systemen sind

- elektrische Größen (Strom, Spannung, Feldstärke, magnetische Flussdichte usw.)
- mechanische Größen (Weg, Geschwindigkeit, Beschleunigung, Kraft, Drehmoment, Temperatur, Druck usw.).

Ein wesentliches Merkmal mechatronischer Systeme besteht darin, dass ihre Eigenschaften in hohem Maße durch nichtmaterielle Elemente, d.h. durch Software, bestimmt werden. Die Verarbeitung der Prozessdaten erfolgt durch speziell für die Echtzeitverarbeitung geeignete Mikrocontroller. Sie enthalten die dazu notwendigen Funktionen, wie Datenspeicher, Programmspeicher, AD-Wandler, I/O-Ports, Interruptverwaltung usw.

Die Prozessdatenverarbeitung geschieht in mehreren Ebenen und übernimmt - je nach Ausbaustufe - verschiedene Aufgaben der Regelung, Überwachung und Optimierung.

1.1.3 Hintergründe und Motivation

Das Zusammenwachsen von Mechanik, Elektronik und Informatik unter dem Dach der Mechatronik hat in den letzten Jahren zu einer großen Zahl wissenschaftlich sehr interessanter und wirtschaftlich sehr erfolgreicher Produkte geführt. Für wichtige Schlüsselbranchen der

Welt, wie die Automobil- und Verkehrstechnik, die Werkzeugmaschinen insbesondere die Robotik oder z.B. die Medizintechnik, wirkt die Mechatronik als Innovationsmotor für zukünftige Produktentwicklungen. Die Integration neuer Funktionen in ein Produkt über ursprünglich bestehenden Fachbereichsgrenzen hinweg, legt dabei völlig neue Freiheitsgrade offen, welche im Rahmen der Entwicklung mechatronischer Produkt genutzt werden können aber auch genutzt werden müssen, um die geforderten Produkteigenschaften darzustellen. Nur das sorgfältig aufeinander abgestimmte Zusammenspiel aller Einzelfunktionen, -bauteile und -baugruppen innerhalb eines mechatronischen Gesamtsystems garantiert ein optimales Gesamtergebnis [GAU].

Allerdings stellt dieser Integrationsprozess traditionell sehr unterschiedlich geprägter Domänen wie *Mechanik/Konstruktion*, *Elektronik/Electronic Design* und *Informatik/Software Design* bis heute eine Herausforderung sowohl für die Unternehmen als auch für die Wissenschaft dar. Die Antwort auf diese Herausforderungen der „Mechatronisierung“ besteht in einem Modulkonzept, wobei das Gesamtprodukt (z.B. in der Automobilindustrie das Fahrzeug) in einem hierarchischen Ansatz auf relativ autarke und somit in ihrer Komplexität begrenzte Module bzw. Modulgruppen herunter gebrochen wird. Module bilden eine *geometrisch/räumliche* und *funktionale* Kapsel mit wohl definierten Schnittstellen, so dass Konzeption, Entwicklung, Fertigung, Montage und Test eines Moduls mit den geringsten möglichen Wechselwirkungen zu anderen Modulen erfolgen kann. Diese Entkopplungswirkung der Module (zusammen mit anderen wichtigen Eigenschaften wie Standardisierbarkeit u.a.) sichert die Basis der heutigen Entwicklungs- und Produktionsprozesse [BER].

Allerdings kann die Entkopplung keinesfalls vollständig sein, sondern muss durch eine geschickte Wahl der Modulgrenzen und durch eine konsequente Werkzeugunterstützung insbesondere der Spezifikations- und Entwicklungsprozesse angenähert werden. Eine leistungsfähige durchgängig werkzeugunterstützte Entwicklungsmethodik stellt dabei eine der wichtigsten Voraussetzungen für das erfolgreiche Abschneiden der Unternehmen im internationalen Wettbewerb dar.

1.2 Stand der Wissenschaft und Technik

Eine durchgängige, alle Domänen umfassende entwicklungsbegleitende Simulation wird allgemein als eine der wichtigsten methodischen Grundlagen moderner Entwicklungsprozesse gesehen. Dabei unterstützen funktionale Produktmodelle/Verhaltensmodelle die Konzeption und den optimalen Entwurf mechatronischer Komponenten genau so wie ihre Integration zu komplexen Baugruppen und Gesamtsystemen [HAH98], [BER], [BER01], [ADM]. Aufgrund der großen wissenschaftlichen und wirtschaftlichen Bedeutung gibt es auf dem Gebiet der Modellierung mechanischer und elektrischer Systeme eine große Zahl wichtiger Methoden und Werkzeuge, die eine Erwähnung verdient hätten.

1.2.1 Modellierung elektrischer Systeme

Die Modellierung elektrischer Systeme erfolgt in einem stark standardisierten Rahmen. Durch die frühe Festlegung auf den Modellierungsstandard SPICE [PSP], konnten sich schon früh leistungsfähige Simulationswerkzeuge, wie MENTOR, MULTISIM, u.a. entwickeln. Die engen Grenzen des SPICE Standards verhinderten allerdings die Einbindung von universellen Verhaltensmodellen oder von nicht-elektrischen Modellen aus anderen Domänen. Abhilfe schaffte erst ein neuer Sprachstandard VHDL-AMS, welcher eine sehr flexible Modellierung auf der Basis von algebraischen und Differentialgleichungen ermöglicht. Auch hier stehen zwischenzeitlich leistungsfähige Werkzeuge wie MENTOR, SABER, u.a. zur Verfügung, die aber bei weitem noch nicht den Verbreitungsgrad gefunden haben wie ihre SPICE Vorgänger.

Das größte Problem dieser Werkzeuge bei der Modellierung und Simulation mechatronischer Systeme liegt darin, dass sie im Kern für rein elektrische Netzwerke entwickelt wurden und im Zusammenspiel mit nicht-elektrischen Systemen untragbar große Rechenzeiten und numerische Instabilitäten entstehen können. Um diese Problem zu umgehen, gibt es im Bereich der Leistungselektronik spezielle Lösungen wie SIMPLORER [SIM], PSIM [PSI], CASPOC [CAS] oder PLECS, welche durch geschickte Anpassungen der Modelle und der Simulationsverfahren sehr kurze Simulationszeiten bei ausreichender Genauigkeit ermöglichen. So modelliert beispielsweise PLECS (Piece-wise Linear Electrical Circuit) [ALI99], [PLE] elektrische Bauelemente wie Transistoren und Dioden als ideale Schalter und generiert stückweise lineare Modelle für elektrische Schaltungen.

Interessante Erweiterungen von Netzwerksimulatoren speziell für Mikrosysteme wurden in [BEC97], [CLA03] vorgestellt. Einige dieser Tools unterstützen die gekoppelte Betrachtung elektrischer und mechanischer Systemteile. Die Modellierung komplexer 3-dimensionaler mechanischer Systemteile auf Mehrkörperebene ist mit diesen Tools jedoch nicht möglich.

1.2.2 Modellierung elektro-mechanisch gekoppelter Systeme

Für die dedizierte Modellierung und Simulation elektro-mechanisch gekoppelter Systeme steht eine wesentlich kleinere Zahl von Lösungen zur Verfügung von denen nur zwei Gruppen angesprochen werden sollen. Zur ersten, numerisch ausgerichteten Gruppe gehört ALASKA [ENG04], [ENM], [MAI03], [ENG03], das in der Tradition vieler mechanischer Tools einen leistungsfähigen Modellierungsansatz auf der Basis von verallgemeinerten Lagrange Funktionen bereitstellt. Damit können mechanisch-elektrisch gekoppelte Probleme prinzipiell sehr gut behandelt werden. Es ist jedoch weniger geeignet für typische elektrische Netzwerkstrukturen mit einer größeren Anzahl von schaltenden Elementen aufgrund der stark unterschiedlichen Systemzeitkonstanten und von Unstetigkeiten, die bei der Modellierung als Schalter entstehen.

1.2.3 Modellierungssprachen und symbolverarbeitende Tools

Eine andere Herangehensweise wird durch symbolverarbeitende Tools wie DYMOLA geboten. Mit der objektorientierten Modellierungssprache MODELICA wird eine gut verständliche multidisziplinäre Plattform geboten [TIL00], [HEL02], [OTT03], [MAT98], [MAT98-2].

MODELICA ist eine Sprache zur Modellierung physikalischer Systeme und wurde entwickelt, um gerade auch den Austausch von Modellen und die Entwicklung von Bibliotheken zu unterstützen. MODELICA setzt nicht auf eine ausschließlich kausale Modellierung, bei der Ursache und Wirkung an jedem Bauelement bereits vor der Simulation festgelegt werden müssen. Die Beschreibung der Modelle kann auch in Form echter Gleichungen und nicht nur auf Basis von Zuweisungen erfolgen. MODELICA unterstützt die Beschreibung kontinuierlicher Systeme, die auf Basis differential-algebraischer Gleichungssysteme (DAE)

berechnet werden können. Hinzu kommen Konstrukte, um auch Diskontinuitäten zu erfassen, die etwa bei mechanischen Anschlägen oder Haft-Gleitreibungsübergängen auftreten. Die Möglichkeiten mit den Diskontinuitäten auch ereignisorientierte Vorgänge, z.B. die Übergänge in einem Zustandsgraphen oder die Bewegung von Marken in einem Petrinetz zu beschreiben, ist zwar prinzipiell gegeben, aber in ihren Möglichkeiten durch die üblicherweise unterliegenden Gleichungslöser begrenzt [PEL01].

Sowohl eine strukturelle als auch eine verhaltensorientierte Modellierung ist möglich. Ein besonders hervorstechendes Merkmal von MODELICA ist die Objektorientierung, die hier unter anderem dazu eingesetzt wird, um beispielsweise ein Modell - d.h. genauer gesagt eine Modellklasse - einmal zu deklarieren und mehrfach zu instantiieren, wobei sich bestimmte Parameter für jede Instanz einzeln einstellen lassen. Ähnliche Konzepte gibt es auch in Hardwarebeschreibungssprachen, wie etwa VHDL mit den Möglichkeiten der Instanziierung und Konfiguration. MODELICA bietet darüber hinaus die Möglichkeit der Vererbung zwischen Modellklassen, so dass sich komplexere Modellklassen leicht auf einfachere zurückführen lassen [PEL01].

Die symbolische Bereitstellung der Systemgleichungen ermöglicht ein sehr komfortables und strukturiertes Arbeiten. Allerdings stößt die Symbolverarbeitung bei wachsender Modellgröße an ein hartes Komplexitätsproblem, da die Anzahl und/oder Größe der generierten Gleichungen mit der Systemordnung sehr stark wächst und letztlich nicht mehr handhabbar ist.

An dieser Stelle soll erwähnt werden, dass für klassische Symbolverarbeitungsprogramme wie MATHEMATICA oder MAPLE entsprechende Toolboxen zur Modellierung mechanischer Starrkörpersysteme und elektrischer Netzwerke existieren, die jedoch in ihrer Anwendbarkeit noch stärkeren Restriktionen unterliegen.

1.2.4 Bondgraphen

Bei den Bondgraphen werden die folgenden Zuordnungen vorgenommen: Spannungen werden üblicherweise mit Kräften/Momenten assoziiert und Spannungsgrößen (effort) genannt; Ströme werden mit Geschwindigkeiten/Winkelgeschwindigkeiten assoziiert

und Flussgrößen (flow) genannt [PEL01], [DAM04]. Die im Bondgraphenansatz verwendeten Elemente lassen sich in Ein-, Zwei- und Dreitore einteilen. Die Eintore sind die so genannten C-, L- und R-Elemente, die in der Elektronik Kondensatoren, Spulen und Widerständen sowie in der Mechanik Federn, Massen und Dämpfern entsprechen siehe Tabelle 1.1.

Tabelle 1.1: Zuordnung von Größen und Elementen in Bondgraphen

Bondgraphen	Elektrotechnik	Mechanik, translatorisch	Mechanik, rotatorisch
Spannungsgröße	Spannung	Kraft	Drehmoment
Flussgröße	Strom	Geschwindigkeit	Winkelgeschwindigkeit
C-Element	Kapazität	Federsteifigkeit	Torsionsfedersteifigkeit
I-Element	Induktivität	Massenträgheit	Trägheitsmoment
R-Element	Widerstand	Dämpfung, translatorisch	Dämpfung, rotatorisch
Übertrager	Transformator	Hebel, Flaschenzug	Getriebe

Hinzu kommen Quellen für Spannungs- und Flussgrößen. Als Zweitore sind Übertrager und Gyrotoren definiert. Erstere übertragen jeweils Spannungs- und Flussgrößen in einem festen oder variablen Verhältnis zueinander; letztere setzen die Spannungsgröße auf der einen Seite mit der Flussgröße auf der anderen Seite ins Verhältnis (und umgekehrt). Übertrager repräsentieren damit beispielsweise Transformatoren, Getriebe oder Hebel für kleine Auslenkungen. Ein Gyrotor könnte etwa einen DC-Motor beschreiben. Die Dreitore schließlich stellen serielle oder parallele Verknüpfungen (s-junction Funktion) dar. Die Ein-, Zwei- und Dreitore werden durch Halbpfeile, so genannte Bonds miteinander verknüpft, die jeweils eine Spannungs- und eine Flussgröße tragen. Die Richtung des Pfeils gibt die Richtung des positiven Leistungsflusses an. Die jeweils geleistete Arbeit ergibt sich aus dem Produkt von Spannungs- und Flussgröße. Neben den Halbpfeilen der Bonds gibt es Verbindungen mit einem vollständigen Pfeil, bei denen entweder die Spannungs- oder die Flussgröße vernachlässigt werden. Diese Verbindungen tragen Information, aber keine Energie.

Die Berechnung von Bondgraphen erfordert zunächst die Aufstellung eines entsprechenden Gleichungssystems, das in aller Regel explizit formuliert ist. Dies bedeutet, dass die Gleichungen die Form von Zuweisungen annehmen, was wiederum eine Betrachtung der Kausalität des Systems erfordert. Für jedes Element sind damit Ursache und Wirkung

festzulegen. Greift man ein beliebiges C-, I- oder R-Element heraus, so stellt sich die Frage, ob nun die Spannungsgröße die Ursache der Flussgröße ist, oder umgekehrt. Beides ist möglich und für beide Fälle existieren Gleichungen, die je nach Bedarf in ein Gleichungssystem eingesetzt werden. Insgesamt gilt es, durchgängige Ketten von Ursache Wirkung-Beziehungen aufzustellen, die sich auf eine entsprechende Sequenz von Zuweisungen abbilden lässt. Dies gelingt bei algebraischen Schleifen nicht, so dass hier zusätzliche Maßnahmen erforderlich sind. Bild 1.2 stellt den Bondgraph einer elektrischen Schaltung dar.

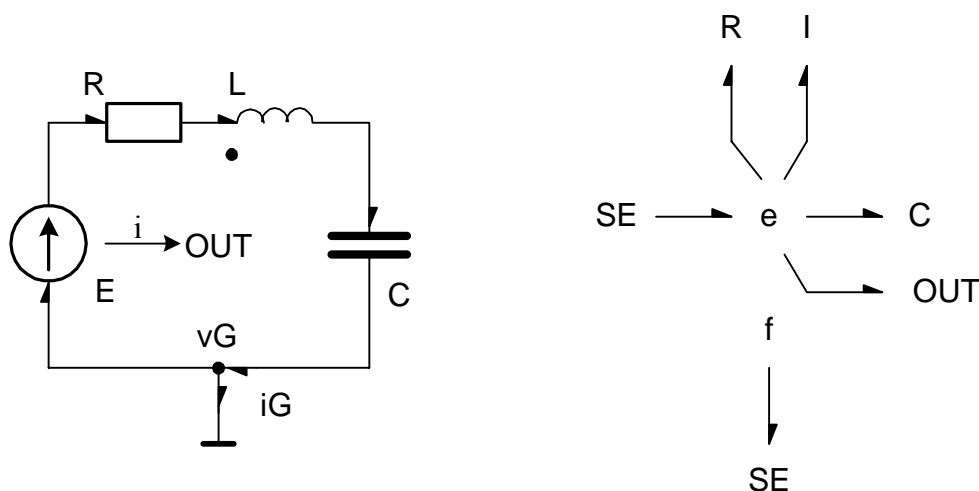


Bild 1.2: Bondgraph eines einfachen elektrischen Systems [DAM04]

Auf der Basis von Bond Graphen wurden sehr leistungsfähige Algorithmen und Tools zur Modellierung und darauf aufbauend auch zur Simulation mechatronischer Systeme entwickelt, welche den Vorzug der gemeinsamen Behandlung unterschiedlicher Domänen bieten, allerdings in Bezug auf die Leistungsfähigkeit und Effizienz den fachspezifischen Verfahren deutlich unterlegen sind.

1.2.5 Blockdiagramme

Daneben gibt es eine Reihe von Werkzeugen, welche ihre Wurzeln nicht in einer spezifischen Domäne sondern in einem Gesamtsystemansatz haben. Hier wäre z.B. das Tool CAMEL zu nennen, welches zusammen mit einer Modellierungs- eine leistungsfähige Analyse- und Entwurfsplattform für mechatronische Systeme bietet [HAH98], [JUN], [HAN00]. Als Erweiterung regelungstechnischer Tools wie z.B. MATLAB/SIMULINK wurde eine Anzahl

von Toolboxen entwickelt, welche die Modellierung mechanischer und elektrischer Systemteile innerhalb eines Blockschaltbilds ermöglichen.

Blockdiagramme stellen ebenso wie Bondgraphen eine Form der strukturellen Modellierung dar. Hierbei ist allerdings in erster Linie die Struktur von Gleichungen gemeint, während sich die Struktur des Systems eher indirekt aus der Struktur des Gleichungssystems ergibt. Blockdiagramme enthalten Blöcke und gerichtete Verbindungen zwischen den Blöcken. Diese Verbindungen beschreiben Signale, die von den Blöcken in andere Signale umgewandelt werden. Das wesentliche Problem hierbei ist, dass die Struktur der Schaltung nicht immer mit der Struktur des Blockdiagramms korrespondiert [PEL01].

Typischerweise weisen diese Toolboxen jedoch immer Einschränkungen auf, so dass sie nicht mit der Leistungsfähigkeit der Mechanik- bzw. Elektronik/Elektrik-Tools mithalten können (siehe PLECS). Daneben gibt es gerade für MATLAB eine große Anzahl von Tool-Kopplungslösungen z.B. zu ADAMS oder zu MENTOR. Diese Lösungen ermöglichen damit immer nur eine einseitige Kopplung der physikalischen (Mechanik- bzw. Elektrik-) Ebene mit der in MATLAB realisierten Verhaltensebene. Dies erfordert in der Regel eine mehrfache Erstellung und Anpassung der verwendeten Modelle für die verschiedenen Tools. Daneben sind die Stabilität und die Performance der Toolkopplung nicht überzeugend.

1.2.7 Komponentenorientierte Modellierung

Die Idee der Nutzung der Komponentenorientierung zum Entwurf mechatronischer Systeme wird in [KAS96], [KAS97], [KAS98], vorgestellt.

Die Grundidee der komponentenorientierten Modellierung elektrischer Systeme wurde in [KAY03] dargestellt. Da die Modellierung einerseits geschlossen und vollständig innerhalb einer Komponente und andererseits für das Gesamtsystem physikalisch richtig erfolgen muss, kommt dem Informationsaustausch zwischen den Komponenten über Schnittstellen eine wichtige Bedeutung zu. Die Schnittstellen sind für bestimmte Arten von Verbindungen standardisiert, auch sie haben somit Entsprechungen in der realen Welt.

Die reale Schnittstelle der Komponenten wird durch ein bidirektionales Wertpaar (U, I) abgebildet:

$$U = \hat{U} + \hat{R} \cdot I \text{ bzw } I = \hat{I} + \hat{G} \cdot U \quad (1.1)$$

Der Strom und die Spannung werden als Summe aus einem stromunabhängigen Potential und einem potentialproportionalen Anteil dargestellt:

Die Datenstruktur der Schnittstelle umfasst zwei Paare mit je zwei Werten: Spannung \hat{U} , Widerstand \hat{R} , Strom \hat{I} und Leitwert \hat{G} . Sie haben virtuellen Charakter. Diese Definition hat Einnschränkungen prinzipieller Art. Die Modellierung vermaschter Systeme ist stark begrenzt. Eine Reihenschaltung zweier Induktivitäten oder Parallelschaltung zweier Kondensatoren ist nicht möglich.

1.3 Ziele und Inhalt der Arbeit

Zusammenfassend kann man feststellen, dass zwar eine sehr breite Methoden- und Werkzeuglandschaft zur Modellierung und Simulation mechanischer und elektrischer – teilweise auch elektromechanischer – Systeme vorhanden ist und auch breit genutzt wird, dass aber keines der Werkzeuge und vor allem nicht die darunter liegenden Methoden – die Basis für eine notwendige durchgängige und domänenübergreifende Unterstützung des Entwicklungsprozesses mechatronischer Systeme und Komponenten bilden kann. Probleme treten insbesondere in folgenden Bereichen auf:

Modulkonzepte

Modulkonzepte sind heute auf der Ebene der Spezifikation mechanischer und elektrischer Systemteile im Entwicklungsprozess fest verankert. Häufig werden diese Modulkonzepte auch während der Modellierung genutzt. Für die Simulation werden diese Module jedoch nicht erhalten sondern in ein flaches mathematisches Problem überführt, was die Weitergabe und Wiederverwendung für Analyse-, Test- und Entwurfszwecke im Entwicklungsprozess stark einschränkt oder verhindert. Diese Methode entspricht der Weitergabe von Software im Quelltext mit allen damit verbundenen bekannten Problemen. Für die Simulation geeignete Module müssen sich wie Software-Module oder Klassen verhalten und in vergleichbarer Form nutzbar sein, indem sie ihren inneren Aufbau kapseln und nach außen eindeutige

Schnittstellen definieren über die sie einfach in übergeordnete Systeme eingebunden werden können.

Single Source

Performanceprobleme sowie nicht durchgängige Modulkonzepte führen letztlich dazu, dass während der Entwicklung mechatronischer Produkte, Modelle von Systemteilen auf sehr unterschiedlichen Modellierungsebenen in sehr unterschiedlichen Modellierungs-Tools teilweise manuell erstellt werden müssen, was prinzipiell fehleranfällig ist und erhebliche Mehrkosten verursacht. Eine enge Anbindung der Simulation z.B. an ein vollständiges 3D-CAD-Modell einschließlich der elektrischen Systemteile ist heute nicht durchgängig möglich, wodurch Innovationspotential bei der Produktentwicklung ungenutzt bleibt und ein großes Risikopotential durch nicht abgesicherte Domänenschnittstellen verbleibt, welches auf elektrische Systeme begrenzt ist.

Performance

Heute vorliegende leistungsfähige Konzepte zur Modellierung bzw. Algorithmen zur Simulation sind entweder für die Behandlung mechanischer Starrkörper- oder elektrischer Netzwerkstrukturen optimiert. Die Behandlung der jeweils „fremden“ Domäne führt zu teils drastischen Performanceverlusten oder ist aufgrund vorliegender Strukturbegrenzungen überhaupt nicht möglich. Weder die abstrakte Behandlung unterschiedlicher Domänen mit Hilfe von DAEs, noch eine einheitliche symbolische Beschreibung über Domänengrenzen hinweg oder gar die Kopplung unterschiedlicher Simulatoren lösen dieses Problem für komplexe mechatronische Produkte.

Eine Lösung dieser Probleme durch die Erforschung, Entwicklung und prototypischen Implementierung einer komponentenorientierten Modellbildungs- und Simulationsmethode verspricht einen wesentlichen Fortschritt der Entwicklungsmethodik mechatronischer Produkte.

Ziel und Inhalt dieser Arbeit bestehen im Aufzeigen der Besonderheiten und in der Herleitung von Lösungsansätzen zur Modellierung elektrischer Teilsysteme im Kontext eines bis zur Code-Ebene konsequent komponentenorientierten computerbasierten Werkzeugs zum

Entwurf und zur Simulation mechatronischer Systeme. In jeweils einem Hauptkapitel der Arbeit werden die dahingehend entwickelten theoretische Grundlagen und spezielle Punkte der komponentenorientierten Modellierung elektrischer Systeme vorgestellt. Ein weiteres Kapitel zeigt beispielhaft die Umsetzung der Ansätze zur Modellierung von Teilen eines konkreten mechatronischen Systems.

Neben dem theoretischen Teil, der Prinzipien und Herangehensweisen zur Inhalt hat, werden auch praktische Umsetzung und bei der Implementierung gemachte Erfahrungen vorgestellt.

2 Theoretische Grundlagen der Methode der komponentenorientierten Modellierung elektrischer Systeme

2.1 Idee der Methode

2.1.1 Darstellung des Systems

Bei der komponentenorientierten Modellierung wird das Gesamtsystem als Menge miteinander verbundener Objekte dargestellt. Mechatronische Objekte wirken mit Hilfe spezieller Schnittstellen zusammen, die realen physikalischen Schnittstellen der Produkte nachgebildet sind (Bild 2.1).

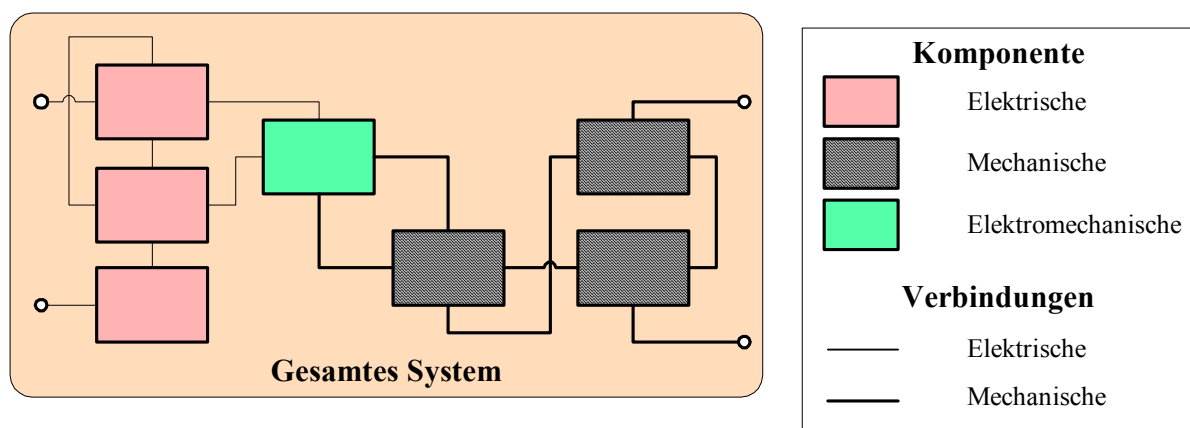


Bild 2.1: Mechatronisches System als Menge verbundener Objekte

Diese Objekte sind elektrische, mechanische oder gemischte Baugruppen. Die Kommunikation zwischen elektrischen Komponenten wird durch elektrische Schnittstellen beschrieben, sowie mechanische Komponenten - durch mechanische Schnittstellen. Elektromechanische (gemischte) Komponenten bestehen aus zwei Modellteilen: einem elektrischen und einem mechanischen.

2.1.2 Definition funktionaler Strukturteile

Der erste Schritt der Modellierungsmethodik besteht in der Definition funktionaler Strukturteile des Systems. Bild 2.2 zeigt exemplarisch wie eine elektrische Struktur auf funktionale Teile verteilt wird.

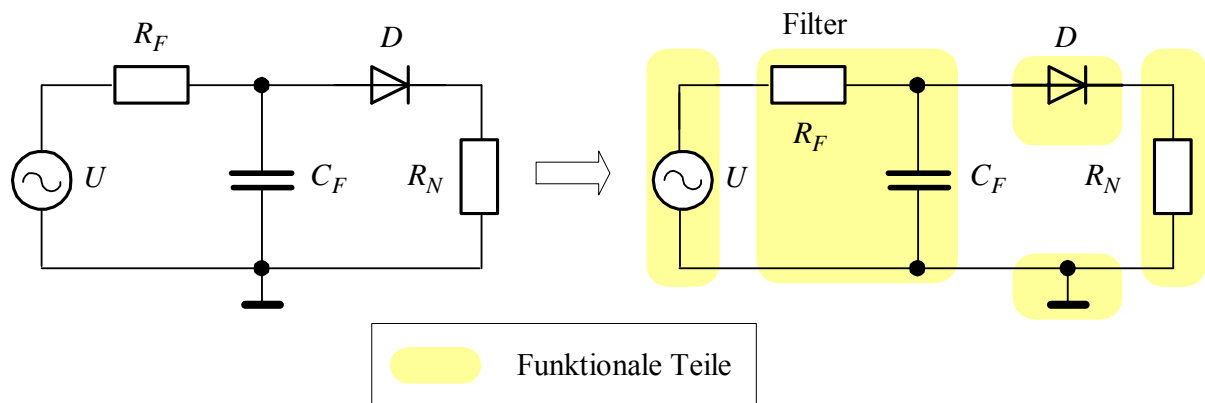


Bild 2.2: Verteilung des Systems auf funktionale Teile

Eine Beispielschaltung des Gleichrichters kann auf die Spannungsquelle U , den *Filter* aus den Elementen R_F , C_F , den *Gleichrichter* aus der Diode D , die *Masse* und die ohmsche Belastung R_N verteilt wird. Diese funktionalen Teile werden später als Struktur- und Grundkomponente definiert werden. Jeder Strukturblock kann aus eigenen Strukturblöcken oder Grundkomponenten bestehen.

2.1.3 Einteilung hierarchischer Ebenen

Aus informationstechnischer Sicht wird das modellierte System auf hierarchische Ebenen aufgeteilt. Die funktionalen Teile der Beispielschaltung in Bild 2.2 werden in 4 Ebenen verteilt (Bild 2.3). Die elementaren Grundkomponenten U , R_F , C_F , D , R_N bilden die niedrigste Ebene 0. Mehrere Grundblöcke und Strukturblöcke der unteren Ebene vereinen sich in die Strukturblöcke der nächsten hierarchischen Ebene. Danach bilden die Elemente R_F , C_F eine Strukturkomponente *Filter* und die Diode D eine Strukturkomponente *Gleichrichter*. Der Gleichrichter kann auch als Grundkomponente gebildet werden. Diese

Komponenten gehören zu Ebene 1. Der Strukturblock Gleichrichterschaltung der Ebene 2 besteht aus den Komponenten Filter und Gleichrichter usw.

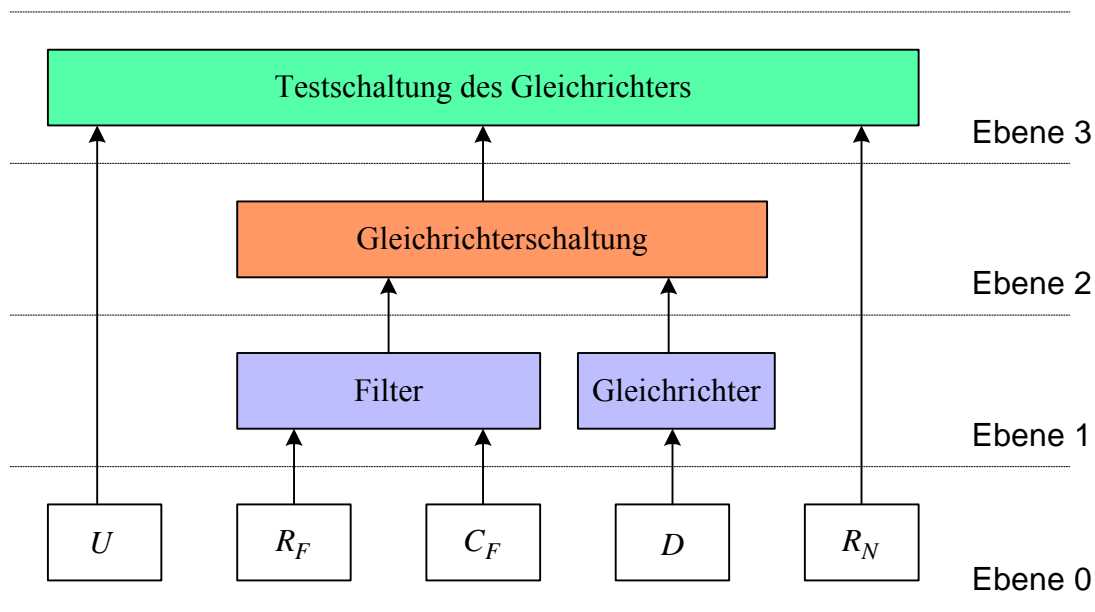


Bild 2.3: Bildung der hierarchischen Ebenen

Der Entwickler trifft die Entscheidung wie die Bauelemente in die funktionalen und hierarchischen Strukturblöcke eingesetzt werden.

2.1.4 Bildung der Komponente

Die Strukturkomponenten können mit zwei Methoden erzeugt werden: entweder aus SPICE ähnlicher Netzliste (oder XML- basierter Netzliste) oder mit Komponenten Abbildung in Visio. Auf der Basis der inneren Struktur der Komponenten und die Definition der elektrischen Schnittstelle werden die Parameter und notwendige algebraische und Differenzialgleichungen generiert. Diese Gleichungen sind von den durch die Schnittstelle fließenden Strömen abhängig. Das System der Gleichungen wird symbolisch vereinfacht und gelöst um die Größe der Matrix und Anzahl der Unbekannten zu reduzieren.

2.1.5 Simulation

Die Komponente der nächsten hierarchischen Ebene bekommt und sammelt die notwendige Information über das Subsystem der vorherigen Ebene via die entwickelte elektrische Schnittstelle. Diese Information wird für die Berechnung der Angaben ihrer eigenen Schnittstelle benutzt. Wenn es möglich ist, errechnet die Komponente lokal die Ströme oder ihre Ableitungen. Auf der höchsten Ebene der Hierarchie werden die restlichen Ströme und ihrer Ableitungen berechnet. Es gibt zwei Möglichkeiten, die Integration der Ableitungen durchzuführen: entweder verteilt auf jeder hierarchischen Ebene, oder zentralisiert auf der höchsten Ebene. Die Komponente sendet die aus den Subblöcken berechneten Ströme oder ihre Ableitungen via die entwickelte Schnittstelle. Auf jeden Fall, werden die inneren Ströme rekursiv in den Subblöcken gerechnet. Auf der Basis der berechneten Ströme werden notwendige Spannungen errechnet.

2.2 Definition der elektrischen Schnittstelle

2.2.1 Allgemeine Annahmen

Zur Beschreibung von Bausteinen realer elektrischer Schaltungen werden so genannte Netzwerkelemente eingeführt; das sind idealisierte Modelle, die einfache mathematische Beziehungen zwischen den auftretenden physikalischen Größen beinhalten [UNB90]. Dabei werden bei den Elementen, die in Bild 2.4 vereinbarten Bezugsrichtungen für die Spannung $U(t)$ und den Strom $I(t)$ gewählt.

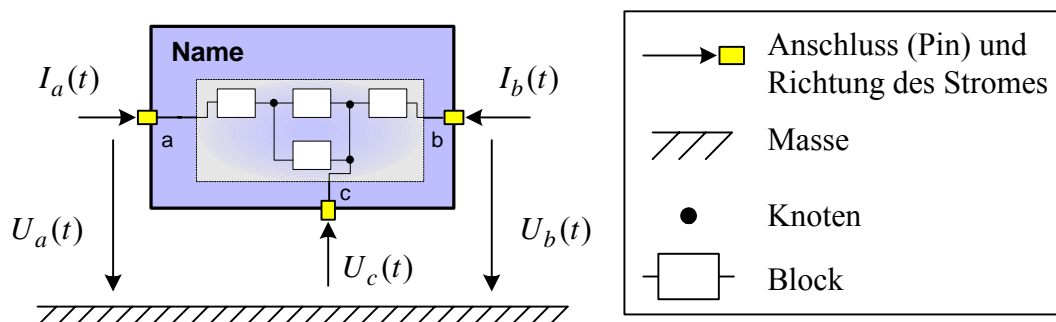


Bild 2.4: Netzwerkelement

Das Element hat einen bestimmten *Namen*. Die innere Struktur des Elementes wird durch die Komponenten und Knoten beschrieben. Die Elemente werden durch die Anschlüsse miteinander verbunden. Anschlüsse werden alphabetisch bezeichnet (a, b, c u.s.w.). Der Pfeil zeigt die Richtung des Stromes, der durch den Anschluss fließt.

Zwischen $U(t)$ und $I(t)$ gibt es für jedes Element eine funktionale Abhängigkeit f , jeweils für seinen Typ des Elementes.

$$U(t) = f[I(t)]. \quad (2.1)$$

Als Variable im Netzwerk treten die Ströme auf, da sämtliche Spannungen durch Ströme ausgedrückt werden. Jedes Element hat eine reale Schnittstelle, die im folgenden als „*elektrischer Pin*“ bezeichnet wird.

Als allgemeine Annahmen sollen dabei gelten:

- Alle Potentiale gelten relativ zu dem als Parameter eingeführten Bezugspotential eines beliebigen Bezugspunktes.
- Das zu simulierende elektrische System besitzt eine Menge von elementaren Zweigströmen, die das System vollständig beschreiben.
- Die Stromrichtung an einem Anschluss wird definiert. Es gilt, dass der Strom an einem Anschluss „in der Komponente“ fließt und wird er mit dem Vorzeichen „+“ verwendet.

2.2.2 Definition der zweipoligen Widerstandskomponente

Eine einfache Grundkomponente Dipol hat zwei Anschlüsse a und b . Die betrachtete Beispielkomponente besteht aus zwei Ohmschen linearen Reihenwiderständen R_1 und R_2 und hat keine Maschen (Bild 2.5). Es wird dabei angenommen, dass durch diese Anschlüsse a und b zwei Eingangsströme I_a, I_b fließen. Die Potentiale U_a, U_b an den Anschlüssen des Blocks gelten als relativ zu dem Bezugspotential U_{REF} definiert. Allgemein kann der Bezugspunkt in einem beliebigen Knoten der Komponente bestimmt werden. Hier wurde er zwischen R_1 und R_2 vordefiniert (Knoten N_1).

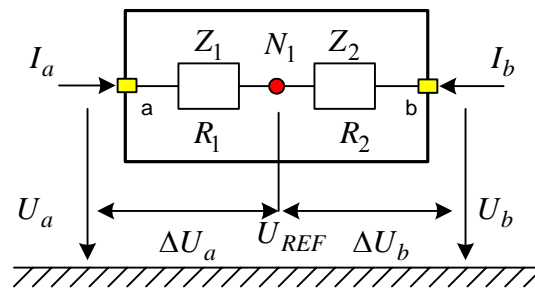


Bild 2.5: Zweipolige Grundkomponente (R-R)

Die Spannungen U_a , U_b in den Anschlüssen können durch das Bezugspotential U_{REF} und eingehende Ströme I_a , I_b ausgedrückt werden:

$$\begin{aligned} U_a &= f_a(U_{REF}, I_a, I_b), \\ U_b &= f_b(U_{REF}, I_a, I_b) \end{aligned} \quad (2.2)$$

oder

$$\begin{aligned} U_a &= U_{REF} + \Delta U_a(I_a, I_b), \\ U_b &= U_{REF} + \Delta U_b(I_a, I_b), \end{aligned} \quad (2.3)$$

Hier sind ΔU_a , ΔU_b Spannungsabfälle über R_1 und R_2 .

Die Knotenregel (1. Kirchhoffsches-Gesetz) für die Eingangsströme lautet:

$$I_a + I_b = 0. \quad (2.4)$$

Bezeichnet man ΔU_a , ΔU_b als die an den linearen stromunabhängigen Widerständen R_1 und R_2 liegende Spannung, so gilt nach dem Ohmschen Gesetz:

$$\begin{aligned} \Delta U_a &= R_1 \cdot I_a, \\ \Delta U_b &= R_2 \cdot I_b. \end{aligned} \quad (2.5)$$

Die betrachtete Komponente hat keine Maschen und der Bezugsknoten teilt die Komponente in mehrere stromunabhängige Teile ein (Bild 2.6). Der Spannungsabfall auf jedem Teil wird nur durch den einzigen Eingangsstrom ausgedrückt.

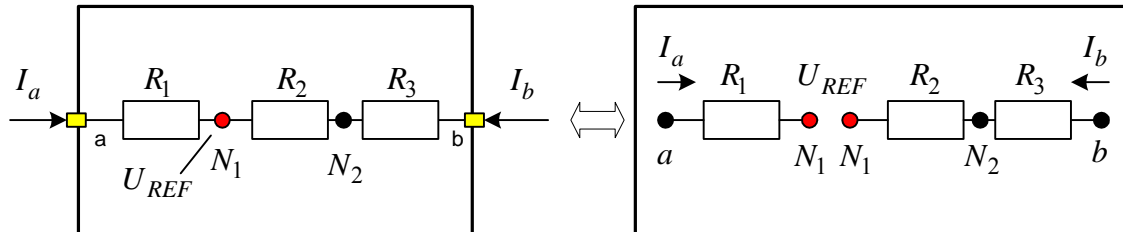


Bild 2.6: Komponente mit innerer Kettenstruktur

Die Spannungen ΔU_a , ΔU_b aus (2.5) kann man als Summe aus einem stromunabhängigen Anteil und einem stromproportionalen Anteil annehmen:

$$\begin{aligned}\Delta U_a &= U_a^* + R_a^* \cdot I_a, \\ \Delta U_b &= U_b^* + R_b^* \cdot I_b.\end{aligned}\quad (2.6)$$

Dabei sind U_a^* , U_b^* die Parameter, die die äquivalente unabhängige Innenspannungsquelle der Komponente beschreiben, falls sie aktiv ist und R_a^* , R_b^* Parameter sind, die äquivalente stromunabhängige Innenwiderstände der Komponente charakterisieren. Beide Parameter definieren die innere Struktur der Komponente [YEF03], [KAS03].

Die Werte der Parameter kann man aus dem Vergleich der Gleichungen (2.5) und (2.6) berechnen:

$$\begin{cases} \Delta U_a(I_a) = U_a^* + R_a^* \cdot I_a \\ \Delta U_a(I_a) = R_1 \cdot I_a \end{cases} \Rightarrow \begin{cases} U_a^* = 0 \text{ und } R_a^* = R_1, \\ \end{cases} \\ \begin{cases} \Delta U_b(I_b) = U_b^* + R_b^* \cdot I_b \\ \Delta U_b(I_b) = R_2 \cdot I_b \end{cases} \Rightarrow \begin{cases} U_b^* = 0 \text{ und } R_b^* = R_2. \end{cases}\end{cases}\quad (2.7)$$

Ohmsche lineare stromunabhängige Komponente in Bild 2.3 kann man auf folgende Weise beschreiben:

Eingangsgröße: I_a, I_b .

Unbekannte: $U_a, U_b, U_{REF}, I_a, I_b$

Parameter der Komponente: $U_a^*, U_b^*, R_a^*, R_b^*$.

Zusammenfassende Gleichungen aus (2.3), (2.4), (2.6):

$$\begin{cases} U_a = U_{REF} + U_a^* + R_a^* \cdot I_a, \\ U_b = U_{REF} + U_b^* + R_b^* \cdot I_b, \\ I_a + I_b = 0. \end{cases} \quad (2.8)$$

Eine Definition der Komponente besteht aus drei Gleichungen (2.8) und fünf Unbekannten. Eine Vordefinition zweier Unbekannter auf der nächsten Ebene lässt dieses System der Gleichungen relativ zu den Strömen I_a, I_b auflösen. Der Einbau der betrachteten Komponente in den geschlossenen Kreis definiert die Spannungen an den Pins a und b (Bild 2.7).

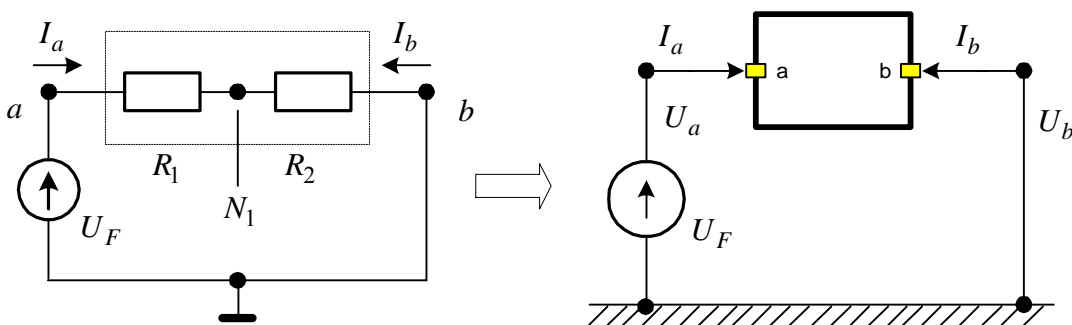


Bild 2.7: Geschlossener Kreis

Hier ist der Anschluss a mit der Spannungsquelle U_F und der Anschluss b mit der Masse verbunden und $U_a = U_F, U_b = 0$.

Das System (2.8) in der Form

$$\begin{cases} U_{REF} + R_1 \cdot I_a = U_F, \\ U_{REF} + R_2 \cdot I_b = 0, \\ I_a = -I_b. \end{cases} \quad (2.9)$$

kann sofort nach I_a , I_b aufgelöst werden:

$$I_a = \frac{U_F}{(R_1 + R_2)}, \quad I_b = -I_a, \quad U_{REF} = R_2 \cdot I_a \quad (2.10)$$

Beispiel 2.1: Definition der Parameter U^* , R^* des Blocks ohne Maschen

Man findet jetzt U^* , R^* - Parameter des Kreises in Bild 2.8.

Der betrachtete Block P besteht aus 3 Basiselementen (Spannungsquelle U_{F1} , zwei Widerständen R_1 , R_2) und 2 Hilfselementen (Knoten N_1 , N_2). Durch die Schnittstellen Pa , Pb fließen zwei Schnittstellenströme I_{Pa} , I_{Pb} . Den Bezugspunkt setzt man an den Knoten N_1 .

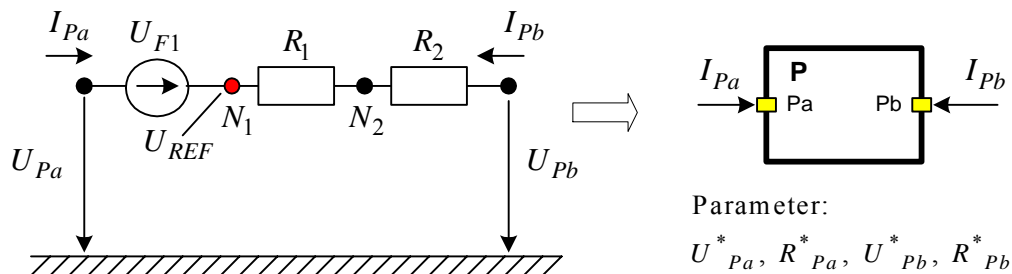


Bild 2.8: Block ohne Maschen

Aus (2.3), (2.4) relativ zu N_1 gilt:

$$\begin{aligned} U_{Pa} &= U_{REF} - U_{F1}, \\ U_{Pb} &= U_{REF} + (R_1 + R_2) \cdot I_{Pb}, \\ I_{Pa} &= -I_{Pb}. \end{aligned} \quad (2.11)$$

Weiterhin folgt aus (2.8) und (2.11):

Für Pa :

$$\begin{cases} U_{Pa} = U_{REF} + U_{Pa}^* + R_{Pa}^* \cdot I_{Pa} \\ U_{Pa} = U_{REF} - U_{F1} \end{cases} \Rightarrow \begin{cases} U_{Pa}^* = -U_{F1} \text{ und } R_{Pa}^* = 0, \end{cases} \quad (2.12)$$

Für Pb :

$$\begin{cases} U_{Pb} = U_{REF} + U_{Pb}^* + R_{Pb}^* \cdot I_{Pb} \\ U_{Pb} = U_{REF} + (R_1 + R_2) \cdot I_{Pb} \end{cases} \Rightarrow \begin{cases} U_{Pb}^* = 0 \text{ und } R_{Pb}^* = R_1 + R_2. \end{cases}$$

Die Parameter U^* , R^* für den Block P sind definiert.

Die Gleichungen (2.8) für den beliebigen Pin v des Blocks ohne Maschen kann man schreiben als:

$$U_v = U_{REF} + U_v^* + R_v^* \cdot I_v. \quad (2.13)$$

Falls eine Komponente eine oder mehrere Maschen oder stromabhängige Elemente hat und das Bezugspotential in der Masche definiert ist, dann hängt U_v für jeden Pin von beiden Schnittstellenströmen I_a , I_b ab:

$$U_v = U_{REF} + U_v^* + R_{va}^* \cdot I_a + R_{vb}^* \cdot I_b. \quad (2.14)$$

Dabei sind R_{va}^* , R_{vb}^* die Parameter, die äquivalente Innenwiderstände des Zweiges v der Komponente in Abhängigkeit von den Strömen I_a , I_b beschreiben.

Beispiel 2.2: Definition der Parameter U^* , R^* der Komponente mit einer Masche

Der Block P in Bild 2.9 besteht aus drei Widerständen R_1 , R_2 , R_3 und zwei Knoten N_1 , N_2 . Durch die Schnittstellen Pa , Pb fließen zwei Schnittstellenströme I_{Pa} , I_{Pb} . Die Widerstände R_1 , R_2 ergeben eine Masche. Der Bezugsknoten ist N_2 .

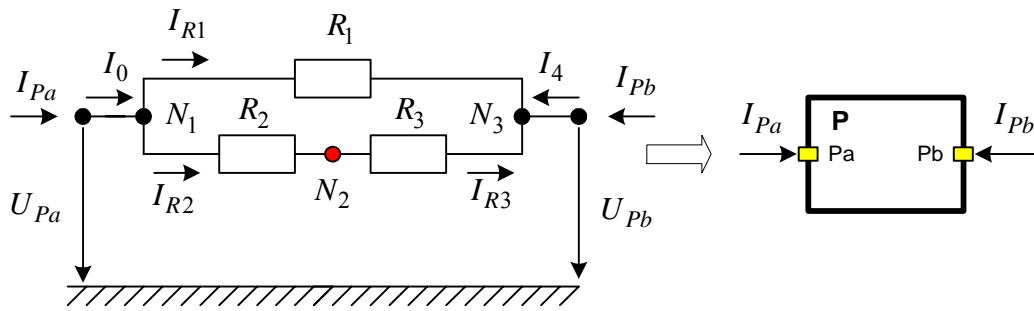


Bild 2.9: Komponente mit einer Masche

Die Knotenregel:

$$\begin{aligned}
 I_{Pa} &= I_0, \\
 I_0 &= I_{R1} + I_{R2}, \\
 I_{R2} &= I_{R3}, \\
 I_{Pb} &= I_4, \\
 I_{Pa} &= -I_{Pb}.
 \end{aligned} \tag{2.15}$$

Die Strom- Spannungsbeziehungen:

$$\begin{aligned}
 U_{Pa} &= U_{N1}, \\
 U_{N3} &= U_{N1} + R_1 \cdot I_{R1}, \\
 U_{N2} &= U_{N1} + R_2 \cdot I_{R2}, \\
 U_{N3} &= U_{N2} + R_3 \cdot I_{R3}, \\
 U_{Pb} &= U_{N3}.
 \end{aligned} \tag{2.16}$$

Man setzt (2.15) und (2.16) zusammen und überführt sie in die Form (2.14). Man kann die Parameter U^* , R^* finden. Falls der Bezugsknoten an den Knoten N_1 eingesetzt wird ($U_{REF} = U_{N1}$), folgt:

Für Pa :

$$\begin{cases}
 U_{Pa} = U_{REF} + U^*_{Pa} + R^*_{Paa} \cdot I_{Pa} + R^*_{Pab} \cdot I_{Pb}, \\
 U_{Pa} = U_{N1}.
 \end{cases} \tag{2.17}$$

$$\Rightarrow \left\{ U^*_{Pa} = 0 \text{ und } R^*_{Paa} = 0, R^*_{Pab} = 0. \right.$$

Für Pb :

$$\begin{cases} U_{Pb} = U_{REF} + U^*_{Pb} + R^*_{Pba} \cdot I_{Pa} + R^*_{Pbb} \cdot I_{Pb}, \\ U_{Pb} = U_{N1} + \frac{R_1(R_2 + R_3)}{R_1 + R_2 + R_3} I_{Pb}. \end{cases}$$

$$\Rightarrow \left\{ U^*_{Pb} = 0 \text{ und } R^*_{Pba} = 0, R^*_{Pbb} = \frac{R_1(R_2 + R_3)}{R_1 + R_2 + R_3} \right\}.$$

Wenn der Bezugsknoten N_2 ist (in der Masche $R_1 \rightarrow R_2 \rightarrow R_3$, wobei $U_{REF} = U_{N2}$):

Für Pa :

$$\begin{cases} U_{Pa} = U_{REF} + U^*_{Pa} + R^*_{Paa} \cdot I_{Pa} + R^*_{Pab} \cdot I_{Pb}, \\ U_{Pa} = U_{N2} + \frac{R_2(R_1 + R_3)}{R_1 + R_2 + R_3} I_{Pa} + \frac{R_2 R_3}{R_1 + R_2 + R_3} I_{Pb}. \end{cases}$$

$$\Rightarrow \left\{ U^*_{Pa} = 0 \text{ und } R^*_{Paa} = \frac{R_2(R_1 + R_3)}{R_1 + R_2 + R_3}, R^*_{Pab} = \frac{R_2 R_3}{R_1 + R_2 + R_3} \right\}.$$

(2.18)

Für Pb :

$$\begin{cases} U_{Pb} = U_{REF} + U^*_{Pb} + R^*_{Pba} \cdot I_{Pa} + R^*_{Pbb} \cdot I_{Pb}, \\ U_{Pb} = U_{N2} + \frac{R_2 R_3}{R_1 + R_2 + R_3} I_{Pa} + \frac{R_3(R_1 + R_2)}{R_1 + R_2 + R_3} I_{Pb}. \end{cases}$$

$$\Rightarrow \left\{ U^*_{Pb} = 0 \text{ und } R^*_{Pba} = \frac{R_2 R_3}{R_1 + R_2 + R_3}, R^*_{Pbb} = \frac{R_3(R_1 + R_2)}{R_1 + R_2 + R_3} \right\}.$$

Die Definierung des Bezugspotentials an beliebigen Knoten ergibt unterschiedliche Werte der Parameter (Tabelle 2.1).

Tabelle 2.1: Parameter des Kreises aus Bild 2.9

Bezugsknoten	U^*	R^*_{Paa}	R^*_{Pab}	R^*_{Pba}	R^*_{Pbb}
N_1	0	0	0	0	$\frac{R_1(R_2 + R_3)}{R_1 + R_2 + R_3}$
N_2	0	$\frac{R_2(R_1 + R_3)}{R_1 + R_2 + R_3}$	$\frac{R_2 R_3}{R_1 + R_2 + R_3}$	$\frac{R_2 R_3}{R_1 + R_2 + R_3}$	$\frac{R_3(R_1 + R_2)}{R_1 + R_2 + R_3}$

Die Richtige Wahl des Bezugspunktes kann den Ausdruck der Parameter vereinfachen. Bei Definition des Bezugspunktes in N_1 lässt sich die Anzahl der Parameter, die nicht gleich Null sind, von 4 auf 1 reduzieren. Es lohnt sich das Bezugspotential dort zu wählen, wo der Bezugsknoten die Komponente auf mehrere stromunabhängige Teile einteilt (Bild 2.6 stellt diesen Fall dar).

2.2.3 Definition der zweipoligen reaktiven Komponente

Das Bild 2.10 zeigt eine Komponente mit den Widerständen R_1, R_2 und zwei Induktivitäten L_1, L_2 .

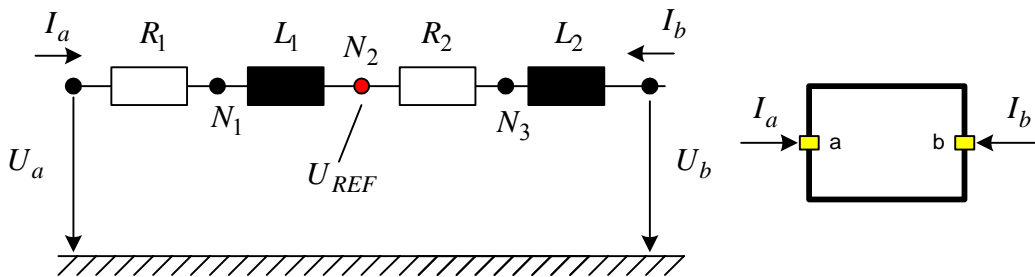


Bild 2.10: Beispiel der reaktiven Komponente

Die Spannungen an den Anschlüssen a, b :

$$\begin{aligned} U_a &= f_a(U_{REF}, I_a, \dot{I}_a, I_b, \dot{I}_b), \\ U_b &= f_b(U_{REF}, I_a, \dot{I}_a, I_b, \dot{I}_b). \end{aligned} \quad (2.19)$$

Nach dem Ohmschen Gesetz für die lineare Komponente:

$$\begin{aligned} U_a &= U_{REF} + \Delta U_a(I_a, \dot{I}_a, I_b, \dot{I}_b), \\ U_b &= U_{REF} + \Delta U_b(I_a, \dot{I}_a, I_b, \dot{I}_b). \end{aligned} \quad (2.20)$$

$$\begin{aligned} U_a &= U_{REF} + R_1 \cdot I_a + L_1 \cdot \dot{I}_a, \\ U_b &= U_{REF} + R_2 \cdot I_b + L_2 \cdot \dot{I}_b. \end{aligned}$$

Für die Ströme gilt:

$$\begin{aligned} I_a + I_b &= 0, \\ \dot{I}_a + \dot{I}_b &= 0. \end{aligned} \quad (2.21)$$

Die Gleichungen (2.14) werden mit dem Anteil, der die lineare Inneninduktivität beschreibt, erweitert:

$$U_v = U_{REF} + U_v^* + R_{va}^* \cdot I_a + R_{vb}^* \cdot I_b + L_{va}^* \cdot \dot{I}_a + L_{vb}^* \cdot \dot{I}_b. \quad (2.22)$$

Dabei sind L_{va}^* , L_{vb}^* die Parameter, die die Induktanz des Zweiges v abhängig von den Strömen I_a , I_b beschreiben.

Weiterhin folgt aus (2.22) für jeden Pin des Blocks in Bild 2.10:

$$\begin{aligned} U_a &= U_{REF} + U_a^* + R_{aa}^* \cdot I_a + R_{ab}^* \cdot I_b + L_{aa}^* \cdot \dot{I}_a + L_{ab}^* \cdot \dot{I}_b, \\ U_b &= U_{REF} + U_b^* + R_{ba}^* \cdot I_a + R_{bb}^* \cdot I_b + L_{ba}^* \cdot \dot{I}_a + L_{bb}^* \cdot \dot{I}_b, \\ I_a + I_b &= 0, \\ \dot{I}_a + \dot{I}_b &= 0. \end{aligned} \quad (2.23)$$

Um die Parameter U^* , R^* , L^* zu ermitteln muss man das System der Differenzialgleichungen symbolisch lösen. Das bereitet Schwierigkeiten, wenn die betrachtete Komponente einige Maschen hat. Im Raum der Laplace-Transformation kann man die Parameter U^* , R^* , L^* durch die Lösung des Systems der algebraischen Gleichungen auswerten.

Die Laplace-Transformation der $I(t)$:

$$L\{I(t)\} \rightarrow I(s), \quad L\{\dot{I}(t)\} \rightarrow s \cdot I(s) - L \cdot I(0). \quad (2.24)$$

Dabei ist $I(0)$ der Anfangswert des Stromes $I(t)$ im Zeitpunkt $t = 0$ [JUS94].

Man verwendet aber die Laplace-Transformation nicht bei der Simulation.

Die Gleichung (2.22) nach der Laplace-Transformation lautet:

$$U_v = U_{REF} + U_v^* + (R_{va}^* + L_{va}^* \cdot s)I_a + (R_{vb}^* + L_{vb}^* \cdot s)I_b - L_{va}^* I_a(0) - L_{vb}^* I_b(0). \quad (2.25)$$

In dieser und in folgenden Formel geht es um die Funktion im Bildbereich, wenn ein s in der Gleichung steht. Die Bildfunktionen $U(s)$, $I(s)$ werden nur als U , I geschrieben um die Ausdrücke abzukürzen. Beim Übergang in den Zeitbereich wird extra drauf hingewiesen.

Man nimmt zunächst an, dass die Anfangswerte der Ströme gleich null sind. Die Gleichungen (2.23) werden dann mit Hilfe (2.24) beschrieben:

$$\begin{aligned} U_a &= U_{REF} + U_a^* + (R_{aa}^* + L_{aa}^* \cdot s) \cdot I_a + (R_{ab}^* + L_{ab}^* \cdot s) \cdot I_b, \\ U_b &= U_{REF} + U_b^* + (R_{ba}^* + L_{ba}^* \cdot s) \cdot I_a + (R_{bb}^* + L_{bb}^* \cdot s) \cdot I_b. \end{aligned} \quad (2.26)$$

Der Fall, wenn die Anfangswerte der Ströme nicht gleich null sind wird später betrachtet.

Man betrachtet die Gleichungen (2.20) und (2.26) gemeinsam. Für die Struktur in Bild 2.10 gibt es eine einfache Lösung:

Für den Pin a :

$$\begin{cases} U_a^* = 0, \\ R_{aa}^* = R_1, R_{ab}^* = 0, \\ L_{aa}^* = L_1, L_{ab}^* = 0. \end{cases}$$

Für den Pin b :

$$\begin{cases} U_b^* = 0, \\ R_{ba}^* = 0, R_{bb}^* = R_2, \\ L_{ba}^* = 0, L_{bb}^* = L_2. \end{cases} \quad (2.27)$$

Nachdem die Parameter gefunden wurden, wendet man die Laplace-Rücktransformation

$$L^{-1}\{I(s)\} \rightarrow I(t), \quad L^{-1}\{s \cdot I(s) - L \cdot I(0)\} \rightarrow \dot{I}(t). \quad (2.28)$$

an und führt in den Zeitbereich über:

$$\begin{aligned} U_a &= U_{REF} + (R_1) \cdot I_a + (L_1) \cdot \dot{I}_a, \\ U_b &= U_{REF} + (R_2) \cdot I_b + (L_2) \cdot \dot{I}_b. \end{aligned} \quad (2.29)$$

2.2.4 Strukturkomponente mit mehreren Eingängen

Ein Strukturblock ist eine Komponente, die aus mehreren Grundkomponenten oder anderen Strukturblöcken besteht. In der Regel haben die Strukturblöcke eine komplizierte Topologie. Ein Strukturblock kann mehrere Maschen haben, deshalb kann die Spannung an einem beliebigen Pin durch mehrere Schnittstellenströme ausgedrückt werden. Die Komponente mit mehreren Pins braucht mehrere Ströme für die Definition. Bild 2.11 zeigt einen Strukturblock mit p Pins (a, b, \dots, p).

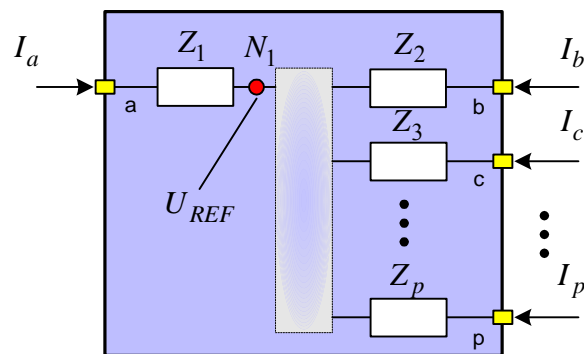


Bild 2.11: Strukturblock mit mehreren Pins

Die Gleichungen (2.19), (2.21), (2.25) können auf mehrere Pins erweitert werden.

Die Spannung U_v an einem beliebigen Pin v ergibt sich zu

$$U_v = f_v(U_{REF}, I_a, I_b, \dots, I_p, \dot{I}_a, \dot{I}_b, \dots, \dot{I}_p) \quad (2.30)$$

und

$$U_v = U_{REF} + U_v^* + (R_{va}^* + L_{va}^* \cdot s)I_a + (R_{vb}^* + L_{vb}^* \cdot s)I_b + \dots + (R_{vp}^* + L_{vp}^* \cdot s)I_p. \quad (2.31)$$

Dabei ist p die Anzahl der Ströme.

Die Schnittstellenspannungen in der Matrixdarstellung für alle Pins aus (2.31) [KAS04]:

$$\begin{pmatrix} U_a \\ U_b \\ \vdots \\ U_p \end{pmatrix} = \begin{pmatrix} U_{REF} \\ U_{REF} \\ \vdots \\ U_{REF} \end{pmatrix} + \begin{pmatrix} U_a^* \\ U_b^* \\ \vdots \\ U_p^* \end{pmatrix} + \begin{pmatrix} Z_{aa}^* & Z_{ab}^* & \cdots & Z_{ap}^* \\ Z_{ba}^* & Z_{bb}^* & \cdots & Z_{bp}^* \\ \vdots & \vdots & \ddots & \vdots \\ Z_{pa}^* & Z_{pb}^* & \cdots & Z_{pp}^* \end{pmatrix} \cdot \begin{pmatrix} I_a \\ I_b \\ \vdots \\ I_p \end{pmatrix}, \quad (2.32)$$

wobei $Z_{vw}^* = R_{vw}^* + L_{vw}^* s$, $v \in N$, $w \in N$.

Dann wird aus dem Gleichungssystem (2.32) die Matrixgleichung im Bildbereich und im Zeitbereich nach der Laplace-Rücktransformation (2.28):

Im Bildbereich:

$$\mathbf{U} = \mathbf{U}_{REF} + \mathbf{U}^* + \mathbf{R}^* \cdot \mathbf{I} + \mathbf{L}^* \cdot s \cdot \mathbf{I}.$$

Im Zeitbereich:

$$\mathbf{U} = \mathbf{U}_{REF} + \mathbf{U}^* + \mathbf{R}^* \cdot \mathbf{I} + \mathbf{L}^* \cdot \dot{\mathbf{I}}, \quad (2.33)$$

$$\mathbf{U} = (U_a \quad U_b \quad \cdots \quad U_p)^T,$$

$$\mathbf{I} = (I_a \quad I_b \quad \cdots \quad I_p)^T,$$

$$\dot{\mathbf{I}} = (\dot{I}_a \quad \dot{I}_b \quad \cdots \quad \dot{I}_p)^T.$$

Dabei sind \mathbf{U} , \mathbf{I} die Vektoren der Eingangsspannungen und der Eingangsströme.

Für die Vereinfachung und Formalisierung der Beschreibung der Komponente, nimmt man an, dass jede Komponente nur Eingänge hat. Dann gilt für die Ströme

$$\sum_{v \in N} I_v = 0, \quad \sum_{v \in N} \dot{I}_v = 0, \quad (2.34)$$

Dabei erfasst $N = \{a, b, \dots, p\}$ alle Pins der Komponente.

2.3 Modellierung der Grundkomponenten

Die zweipoligen elektrischen Grundkomponenten sind Spannungsquelle V_s , Widerstand R , Spule L , Kondensator C , Schalter S und Diode D . Die Besonderheit der Komponenten besteht in der spezifischen vereinfachten Schnittstellengestaltung. Für eine solche Komponente werden nur ein Eingangsstrom und entsprechende Parameter U^* , R^* , L^* definiert.

2.3.1 Modelle der Widerstandskomponenten

Zu den zweipoligen Widerstandskomponenten kann man einen reinen Widerstand und eine Spannungsquelle hinzufügen.

Widerstand

Die zweipolige passive Komponente „Widerstand“ hat, entsprechend der betrachteten Schnittstelledefinition (s. Kapitel 2.2), zwei Eingangsströme I_{Ra} , I_{Rb} durch die Pins und einen Innenwiderstand R (Bild 2.12).

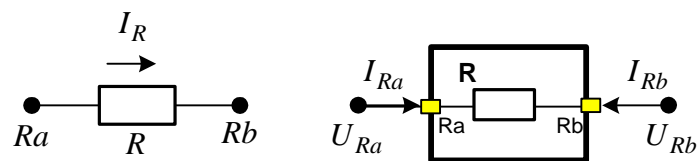


Bild 2.12: Graphische Darstellung der Grundkomponente „Widerstand“

Entsprechend der Gleichung (2.34) gilt

$$I_{Ra} + I_{Rb} = 0. \quad (2.35)$$

Die Ströme I_{Ra} , I_{Rb} sind gleich groß, aber entgegengesetzt gerichtet. Die beiden Ströme können durch einen Strom I_R ersetzt werden:

$$\begin{aligned} I_{Ra} &= I_R, \\ I_{Rb} &= -I_R. \end{aligned} \quad (2.36)$$

Also kann für eine zweipolige Komponente nur ein Strom definiert werden. Durch diese Annahme kann die Anzahl der Eingangsströme, der Gleichungen und der Parameter reduziert werden.

Das Referenzpotential U_{REF} kann am beliebigen Eingangsknoten definiert werden. Setzt man U_{REF} in Rb an, dann kann der gesuchte Parameter relativ zu Rb positiv sein. Bild 2.13 zeigt ein äquivalentes Schema der Komponente „Widerstand“.

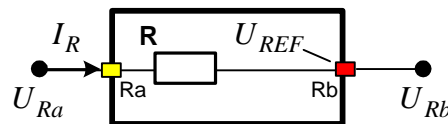


Bild 2.13: Äquivalentes Schema der Grundkomponente „Widerstand“

Die Eingangsspannung U_{Rb} wird automatisch durch U_{REF} errechnet und ist vom Strom I_R unabhängig:

$$U_{Rb} = U_{REF}. \quad (2.37)$$

Die Eingangsspannung U_{Ra} entsprechend die Gleichung (2.32) kann man schreiben als

$$\begin{cases} U_{Ra} = U_{REF} + (U_{Ra}^*) + (Z_{Raa}^*) \cdot I_R, \\ U_{Ra} = U_{REF} + (R) \cdot I_R, \end{cases} \quad (2.38)$$

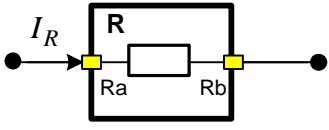
wobei U_{Ra}^* die Spannungsquelle und $Z_{Raa}^* = R_{Raa}^* + L_{Raa}^* \cdot s$ die Impedanz des Zweiges von Eingangspin Ra bis Referenzpunkt Rb ist.

Für den Widerstand in Bild 2.13 ist offenbar:

$$U_{Ra}^* = 0, R_{Raa}^* = R, L_{Raa}^* = 0. \tag{2.39}$$

Die Beschreibung des Modells der Komponente „Widerstand“ ist in Tabelle 2.2 zusammengefasst.

Tabelle 2.2: Beschreibung des Modells des Widerstandes

Graphische Darstellung	Parameter			Anfangswert
		U^*, V	R^*, Ω	
	0	R	0	
Zustandsgröße	Strom durch den Block		Spannungsabfall	
-	I_R		$R \cdot I_R$	

Spannungsquelle

Die zweipolige aktive Komponente „Spannungsquelle“ mit der Spannung U_{VS} und Innenwiderstand R_{VS} ist in Bild 2.14 dargestellt. Durch die Komponente fließt der Eingangsstrom I_{VS} . Der Referenzpunkt wird im Pin VSb gesetzt.

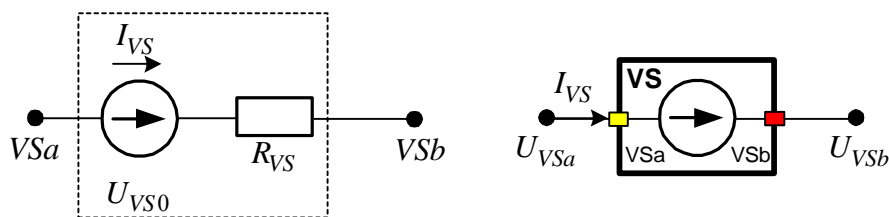


Bild 2.14: Äquivalentes Schema der Grundkomponente „Spannungsquelle“

Die Eingangsspannung U_{VSa} entspricht der Gleichung (2.32):

$$\begin{cases} U_{VSa} = U_{REF} + (U_{VSa}^*) + (R_{VSaa}^* + L_{VSaa}^* s) I_{VS}, \\ U_{VSa} = U_{REF} + (-U_{VS0}) + (R_{VS}) I_{VS}. \end{cases} \tag{2.40}$$

Damit ist

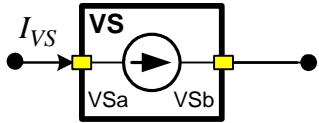
$$U_{VSa}^* = -U_{VS0}, R_{VSaa}^* = R_{VS}, L_{VSaa}^* = 0. \quad (2.41)$$

Die Eingangsspannung U_{VSb} ist vom Strom I_{VS} unabhängig:

$$U_{VSb} = U_{REF}. \quad (2.42)$$

Die Beschreibung des Modells der Komponente „Spannungsquelle“ ist in Tabelle 2.3 zusammengefasst.

Tabelle 2.3: Beschreibung des Modells der Spannungsquelle

Graphische Darstellung	Parameter			Anfangswert
	U^*, V	R^*, Ω	L^*, H	-
	$-U_{VS0}$	R_{VS}	0	
Zustandsgröße	Strom durch den Block		Spannung	
-	I_{VS}		$U_{VS} - R_{VS} \cdot I_{VS}$	

2.3.2 Modelle der Komponente mit den Zuständen

Die reine Spule und der Kondensator sind die reaktiven Grundkomponenten. Die Besonderheiten der idealen Spule und des idealen Kondensators bestehen im Vorhandensein von Zustandsgrößen. Strom bzw. Spannung treten in abgeleiteter Form auf, die mathematische Beschreibung des Systems ist eine Differentialgleichung. Die Gleichung zur Berechnung der Zustandsableitung kann jeweils erst nach Durchlaufen des Stromberechnungspfades erfolgen und bildet somit eine separate Berechnungsstufe.

Spule

Bei der idealen Spule L ist der Eingangsstrom I_L Zustandsgröße und somit zahlenmäßig in jeden Zeitpunkt bekannt. Der Referenzpunkt wird im Pin Lb gesetzt (Bild 2.15).

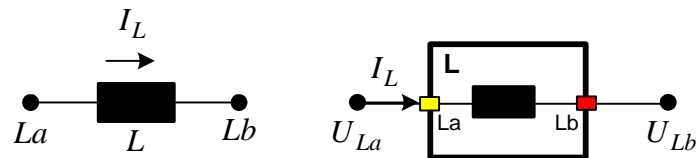


Bild 2.15: Graphische Darstellung der Grundkomponente „Ideale Spule“

Die Eingangsspannung U_{La} aus (2.32) kann geschrieben werden als

$$\begin{cases} U_{La} = U_{REF} + (U_{La}^*) + (L_{Laa}^* s) \cdot I_L - L_{Laa}^* \cdot I_L(0), \\ U_{La} = U_{REF} + (Ls) \cdot I_L - L \cdot (I_L(0)), \end{cases} \quad (2.43)$$

wobei L_{Laa}^* der Parameter der Induktivität des Zweiges vom Eingangspin La bis zum Referenzpunkt Lb ist. $I_L(0)$ der Anfangswert des Stromes in der Spule.

Der Spannungsabfall auf der Spule:

$$U_L = U_{La} - U_{Lb} = L \cdot \dot{I}_L. \quad (2.44)$$

Die Beschreibung des Modells der idealen Spule ist in Tabelle 2.4 dargestellt.

Tabelle 2.4: Beschreibung des Modells der idealen Spule

Graphische Darstellung	Parameter			Anfangswert
	U^*, V	R^*, Ω	L^*, H	$I_L(0)$
	0	0	L	
Zustandsgröße	Strom durch den Block		Spannungsabfall	
I_L	I_L		$L \cdot \dot{I}_L$	

Die reale Spule im Bereich der niedrigen Frequenzen kann man als eine Kombination aus dem Innenwiderstand R_L und der Induktivität L betrachten (Bild 2.16).

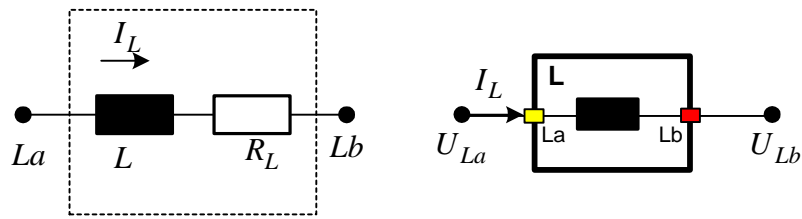


Bild 2.16: Äquivalentes Schema der Grundkomponente „Reale Spule“

Tabelle 2.5 Stellt die Beschreibung der realen Spule dar.

Tabelle 2.5: Beschreibung des Modells der realen Spule

Graphische Darstellung	Parameter			Anfangswert
		U^*, V	R^*, Ω	
	0	R_L	L	
Zustandsgröße	Strom durch den Block		Spannungsabfall	
I_L	I_L		$L \cdot \dot{I}_L + R_L \cdot I_L$	

Kondensator

Der Kondensator C hat eine Kapazität C . Der Eingangsstrom ist I_C , der Referenzpunkt ist der Pin Cb (Bild 2.17).

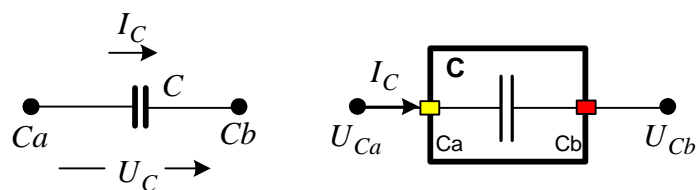


Bild 2.17: Graphische Darstellung der Grundkomponente „Kondensator“

Die Spannung U_C über dem Kondensator hängt vom Strom I_C ab:

$$U_C(I_C) = \frac{1}{C} \int I_C dt. \tag{2.45}$$

U_C ist eine Zustandsgröße. Die wird in der Komponente durch Integration des Stromes I_C errechnet und ist so zahlenmäßig zu jedem Zeitpunkt bekannt.

Die Eingangsspannung U_{Ca} ergibt sich:

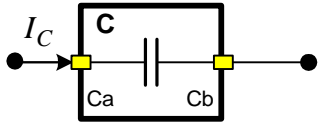
$$\begin{aligned} U_{Ca} &= U_{REF} + (U_{Ca}^*) + (R_{Caa}^* + L_{Caa}^* s) I_C, \\ U_{Ca} &= U_{REF} + (U_C(I_C)). \end{aligned} \quad (2.46)$$

Aus (2.46) gilt

$$U_{Ca}^* = U_C(I_C), \quad R_{Caa}^* = 0, \quad L_{Caa}^* = 0. \quad (2.47)$$

Die Beschreibung des Modells des idealen Kondensators ist in Tabelle 2.6 dargestellt.

Tabelle 2.6: Beschreibung des Modells der Spannungsquelle

Graphische Darstellung	Parameter			Anfangswert
	U^*, V	R^*, Ω	L^*, H	$U_C(0)$
	$U_C(I_C)$	0	0	
Zustandsgröße	Strom durch den Block		Spannung	
U_C	I_C		$U_C(I_C) = \frac{1}{C} \int I_C dt.$	

2.3.3 Modell des MOSFET Transistors und des Schalters

Ein wichtiges Teil des mechatronischen Netzwerkes ist der MOSFET Transistor. In Antrieben oder als Wandler wird üblich ein MOSFET Transistor im Schaltbetrieb verwendet. Ein MOSFET im Schaltbetrieb kann man als von außen gesteuerten idealen Schalter mit zwei Zuständen betrachten (Bild 2.18).

Hier fließt der Strom I_S durch den Zweig $Sa - Sb$. Das Steuersignal ist Str .

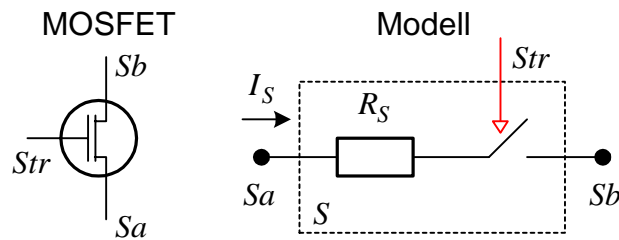


Bild 2.18: MOSFET Transistor und sein Schaltmodell

Im Zustand 1 bleibt der Schalter geschlossen, der Widerstand des Schalters $R_S = R_{Son}$ ist sehr klein, durch die Komponente fließt der Strom $I_S = I_{Son}$ (Bild 2.19).

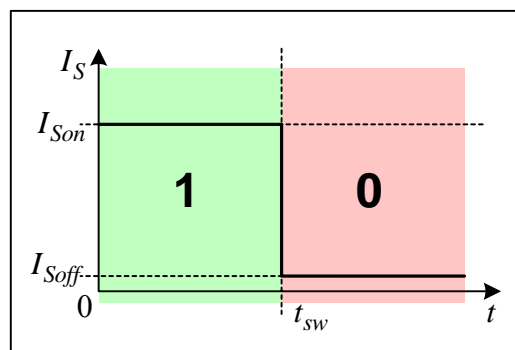


Bild 2.19: Zustände des Schalters

Im Zustand 0 ist der Schalter geöffnet, der Widerstand $R_S = R_{Soff}$ ist sehr groß und der Strom $I_S = I_{Soff}$ ist sehr klein. Die Umschaltung zwischen den Zuständen wird für den idealen Schalter sofort zum Zeitpunkt t_{sw} erfolgen (Falls $R_{Soff} \rightarrow \infty$, dann $I_{Soff} \rightarrow 0$).

Das komponentenorientierte Modell des Schalters kann man wie eine Komponente „Widerstand“ mit zwei möglichen Zuständen betrachten (Bild 2.20).

Hier ist I_S der Eingangsstrom. Der Referenzpunkt wird im Pin Sb gesetzt. Die Steuerung erfolgt durch den Pin Str .

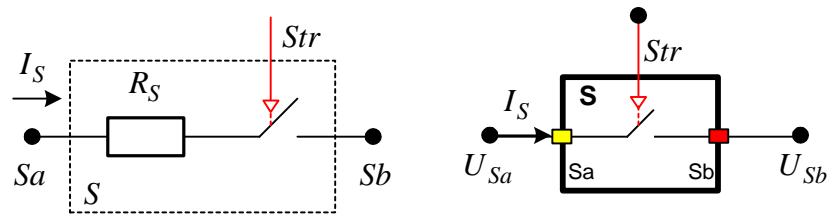


Bild 2.20: Äquivalentes Schema der Grundkomponente „Schalter“

Die Eingangsspannung U_{Sa} für beide Zustände ist:

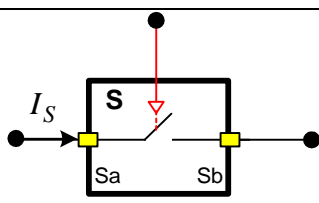
$$\begin{cases} U_{Sa} = U_{REF} + (U^*_{Sa}) + (R^*_{Saa} + L^*_{Saa} s) \cdot I_S, \\ U_{Sa} = U_{REF} + (R_S) \cdot I_S, \end{cases} \quad (2.48)$$

und

$$U^*_{Sa} = 0, R^*_{Saa} = R_S, L^*_{Saa} = 0. \quad (2.49)$$

Die komplette Beschreibung des Schalters für die Zustände 0 und 1 ist in Tabelle 2.7 zusammengefasst.

Tabelle 2.7: Beschreibung des Modells des Schalters

Graphische Darstellung	Parameter			Anfangswert
	U^*, V	R^*, Ω	L^*, H	
				-
	-	R_S	0	
Zustandsgröße	Strom durch den Block		Spannungsabfall	
-	I_S		$R_S \cdot I_S$	

Für beide Zustände gilt wie oben beschrieben:

$$\begin{aligned} \text{Zustand 1: } R_S^* &= R_{Son}, I_S = I_{Son}; \\ \text{Zustand 0: } R_S^* &= R_{Soff}, I_S = I_{Soff}. \end{aligned} \quad (2.50)$$

2.3.5 Modell des Piezoaktors

Die Modellierung des Piezoaktors geht auf Untersuchungen am Dienstleistungszentrum für „Neue Steuer- und Reglersysteme für fluidische Antriebe“ zurück, das bis Ende 1996 am IMAT bestand, sowie drauf aufbauende theoretische und praktische Forschungen [KAS96-2], [KAS98-1], [KAS99].

Das Modell wurde unter dem Aspekt des komponentenorientierten Ansatzes überarbeitet. Dies betrifft vor allem die Gestaltung der mechanischen und elektrischen Schnittstellen und im Zusammenhang damit die Entkopplung der Teilberechnungen Strom/Spannung bzw. Kinetik/Kräfte/Beschleunigung [KAY03].

Bild 2.21 zeigt die Struktur des Simulationsmodells „Piezoaktor“.

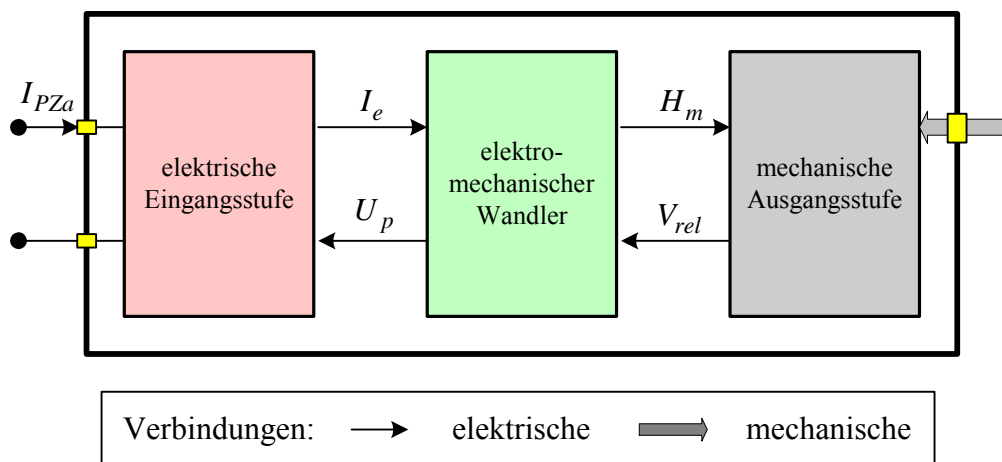


Bild 2.21: Struktur des Simulationsmodells „Piezoaktor“

Das Modell des Piezoaktors besteht aus den drei Teilen:

- Elektrische Eingangsstufe;
- Elektromechanischer Wandler;
- Mechanische Ausgangsstufe.

Die zwischen den Teilen ausgetauschten Größen sind Aktorstrom I_e , Aktorspannung U_p , Aktorhub H_m und Relativgeschwindigkeit V_{rel} der Aktorendflächen. Vorgreifend sei erwähnt, dass der elektromechanische Wandler eine Integrationsstufe (Strom \rightarrow Ladung) im Berechnungspfad seiner Ausgänge enthält. Es gibt daher keinen direkten Durchgriff der Eingänge I_e oder V_{rel} auf die Ausgänge H_m oder U_p , der über die anderen Teile einen Deadlock (wechselseitige Abhängigkeit) verursachen könnte. Der mathematisch-physikalische Hintergrund des elektromechanischen und mechanischen Modells wird in den oben angeführten Literaturstellen dargestellt. Die am Institut durchgeführten Untersuchungen sind auch Basis des entwickelten Modells, auf Details der Implementierung soll daher hier verzichtet werden. Die wichtigste Beschreibung des Aktors ist seine elektrische Ladung Q_{PZ} , die gleichzeitig Zustandvariable ist. Dadurch wird das elektrisch kapazitive Grundverhalten des Aktors bestimmt.

Der elektrische Anteil des Modells wird als elektrische Komponente PZ modelliert (Bild 2.22).

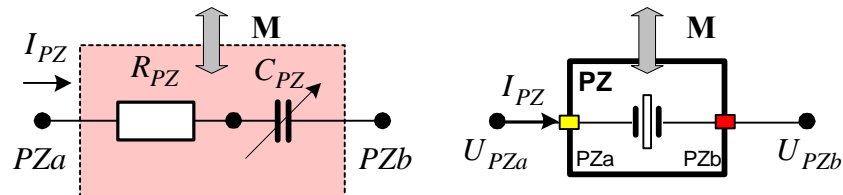


Bild 2.22: Elektrische Eingangsstufe des Modells „Piezoaktor“

Das kapazitive Grundverhalten wird als gesteuerter Kondensator C_{PZ} modelliert. Die Kapazität des Kondensators ist vom mechanischen Verhalten \mathbf{M} abhängig. Die auftretenden ohmschen Verluste können durch einen Reihenwiderstand R_{PZ} modelliert werden. Der Eingangsstrom ist I_{PZ} , Referenzpunkt ist der Pin PZb

Die Spannung $U_{C_{PZ}}$ über dem Kondensator C_{PZ} hängt vom Strom I_{PZ} ab:

$$C_{PZ} = f(\mathbf{M}), \quad (2.51)$$

$$U_{C_{PZ}}(I_{C_{PZ}}) = \frac{1}{C_{PZ}} \int I_{C_{PZ}} dt.$$

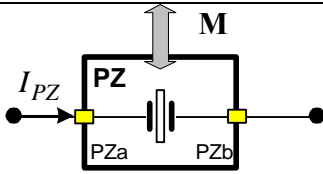
$U_{C_{PZ}}$ wird in der Komponente durch Integration des Stromes I_{PZ} errechnet und ist so zahlenmäßig zu jedem Zeitpunkt bekannt.

Die Eingangsspannung U_{PZa} ergibt sich:

$$\begin{aligned} U_{PZa} &= U_{REF} + (U_{PZa}^*) + (R_{PZaa}^* + L_{PZaa}^* s) I_{PZ}, \\ U_{PZa} &= U_{REF} + (U_{C_{PZ}}) + (R_{PZ}) I_{PZ}. \end{aligned} \quad (2.52)$$

Die Beschreibung des elektrischen Anteils des Modells des Piezoaktors ist in Tabelle 2.8 dargestellt.

Tabelle 2.8: Beschreibung des elektrischen Anteils des Modells des Piezoaktors

Graphische Darstellung	Parameter			Anfangswert
		U^*, V	R^*, Ω	
$U_{C_{PZ}}$		R_{PZ}	0	
Zustandsgröße	Strom durch den Block		Spannung	
$U_{C_{PZ}}$	I_{PZ}		$U_{PZ} = U_{C_{PZ}} + R_{PZ} I_{PZ}$	

2.3.6 Modellierung der Diode

Die Diode ist eine fundamentale Komponente der Halbleitereinheiten. Dioden sind Komponenten mit nichtlinearen Widerstandsverhalten (Bild 2.23).

Es wird ein statisches Modell der Diode betrachtet [DAM03], [KIE94], [MAS93].

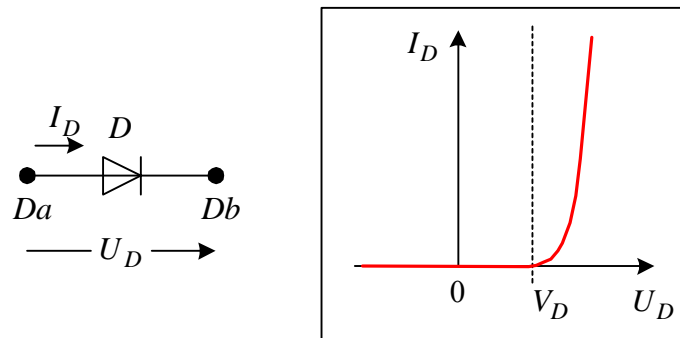


Bild 2.23: Elektrisches Symbol und Kennlinie der Diode

Der Strom durch Diode I_D ist von der Spannung U_D nichtlinear abhängig:

$$I_D = I_S \left(e^{U_D/V_T} - 1 \right). \quad (2.53)$$

Dabei ist I_S der Sättigungssperrstrom. V_T ist die Temperaturspannung:

$$V_T = \frac{kT}{q}, \quad (2.54)$$

wobei $k = 1,3806 \cdot 10^{23} \text{ J} \cdot \text{K}^{-1}$ die Boltzmannkonstante, T die Temperatur in Kelvin (K) und $q = 1,6022 \cdot 10^{-19} \text{ C}$ die Elementarladung ist.

Das vereinfachte Modell der Diode kann man als lineares Ersatzmodell darstellen (Bild 2.24).

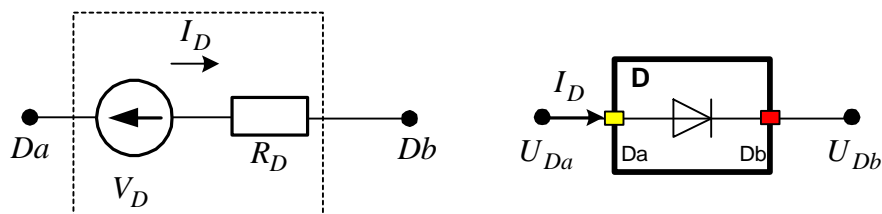


Bild 2.24: Das lineare Ersatzmodell und die komponentenorientierte Darstellung der Diode

Das Modell besteht aus dem differentiellen Widerstand R_D und der Ersatzspannung V_D . Der differentielle Widerstand der Diode ergibt sich aus der Tangente durch den Arbeitspunkt der Diode. Durch die Verwendung einer Geraden anstatt der tatsächlichen Exponentialfunktion werden die benötigten Rechenschritte wesentlich vereinfacht.

Das Modell unterscheidet zwei mögliche Betriebszustände der Diode: Die Diode leitet (Zustand 1, $I_D > I_S$ und $U_D > V_D$, $R_D = R_d$) und die Diode gesperrt (Zustand 0, $I_D < I_S$ und $U_D < V_D$, $R_D = R_S$) (Bild 2.25).

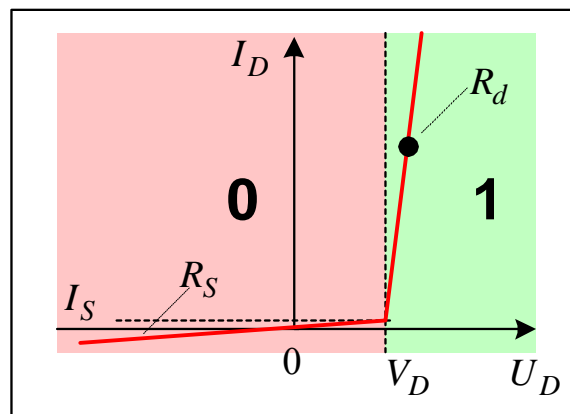


Bild 2.25: Betriebszustände der Diode

Der Strom durch die Diode für beiden Zustände:

$$I_D = \frac{(U_D - V_D)}{R_d}, \quad \text{wenn } I_D > I_S \text{ und } U_D > V_D; \quad (2.55)$$

$$I_D = \frac{U_D}{R_S}, \quad \text{wenn } I_D < I_S \text{ und } U_D < V_D,$$

Dabei ist R_S der Widerstand der Diode im gesperrten Zustand und

$$R_S = \frac{V_D}{I_S}. \quad (2.56)$$

Bei der betrachteten Methode wird der Strom als Eingangsvariable betrachtet. Dann folgt aus (2.55) die Spannung an der Diode:

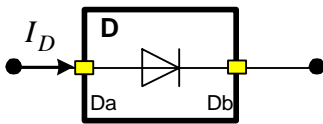
$$\begin{aligned}
 U_D &= V_D + I_D R_d, \text{ wenn } I_D > I_S; \\
 U_D &= I_D R_S, \text{ wenn } I_D < I_S.
 \end{aligned}
 \tag{2.57}$$

Der Vergleich (2.57) mit (2.33) ergibt die Werte der Parameter U^* , R^* , L^* :

$$\begin{aligned}
 U_{Da}^* &= V_O, R_{Daa}^* = R_D, L_{Daa}^* = 0, \\
 \begin{cases} V_O = V_D, R_D = R_d, & \text{wenn Zustand 1;} \\ V_O = 0, R_D = R_S, & \text{wenn Zustand 0.} \end{cases}
 \end{aligned}
 \tag{2.58}$$

Die Beschreibung des linearen Modells der Diode ist in Tabelle 2.9 dargestellt.

Tabelle 2.9: Beschreibung des linearen Modells der Diode

Graphische Darstellung	Parameter			Anfangswert
	U^*, V	R^*, Ω	L^*, H	
	V_O	R_D	0	-
	Zustandsgröße			
-	I_D		$U_D = V_O + I_D R_D$	

2.4 Allgemeine Methodik der Berechnung der Schnittstellenparameter

Das Netzwerk mit n Zweigen hat n unbekannte Ströme, man benötigt also n voneinander unabhängige Gleichungen [LUN87]. Enthält das Netzwerk m Knoten, so sind $m-1$ Knotengleichungen unanständig voneinander.

Solange man auf Bauelemente mit linearen Strom-Spannungsbeziehungen

$$U_z = f(I_z, \dot{I}_z). \tag{2.59}$$

beschränkt, liefern diese ein Gleichungssystem der Form:

$$\mathbf{R} \cdot \mathbf{I} + \mathbf{L} \cdot \dot{\mathbf{I}} = \mathbf{U}_0, \quad (2.60)$$

wobei \mathbf{R} die Widerstandsmatrix, \mathbf{L} die Induktionsmatrix, \mathbf{U}_0 der Vektor der bekannten unabhängigen Spannungsquellen, \mathbf{I} der Vektor der Zweigströme und $\dot{\mathbf{I}}$ dessen Ableitungen nach der Zeit sind. \mathbf{R} und \mathbf{L} sind $n \times n$ Matrizen, \mathbf{U}_0 , \mathbf{I} und $\dot{\mathbf{I}}$ sind Vektoren der Länge n , n ist die Anzahl der Zweige.

Wenn man im Bildbereich arbeitet, kann man die Matrizen \mathbf{R} und \mathbf{L} zu einer einzigen $n \times n$ Matrix \mathbf{Z} zusammenfassen, in der neben den Widerstandswerten R auch Produkte $L \cdot s$ der Induktivitäten L mit der algebraischen Variablen s der Laplace-Transformation auftreten. Man verwendet diese Schreibweise gelegentlich, weil sie eine bequemere Darstellung der Gleichungen ermöglicht, verwendet aber die Laplace-Transformation nicht bei den Rechnungen.

Das Ziel ist es, das Gleichungssystem der analysierte Komponente so aufstellen, dass man die Spannungen U auf den Pins in der Form

$$\mathbf{U} = \mathbf{U}_{\text{REF}} + \mathbf{U}^* + \mathbf{R}^* \cdot \mathbf{I} + \mathbf{L}^* s \cdot \mathbf{I} \quad (2.61)$$

darstellen kann. Wobei \mathbf{I} der Vektor der Eingangsströme durch die Pins und \mathbf{U}_{REF} der Vektor der Bezugsspannungen ist [KAS04].

Das System der Gleichungen in der Form (2.61) hilft die Parameter \mathbf{U}^* , \mathbf{R}^* , \mathbf{L}^* für eine beliebige Netzwerkstruktur der Komponente zu definieren. Dafür wird die Methode der Zweigstromanalyse modifiziert.

2.4.1 Die Inzidenzmatrix eines gerichteten Graphen und Stromgleichungen

Ein elektrisches Netzwerk kann man als einen Graphen betrachten.

Man betrachtet einen offenen Graphen mit m Knoten und n Zweigen, wobei die Zweige jeweils orientiert sind. Eine Menge der Knoten besteht aus Pins und inneren Knoten des Graphen. Ein Zweig beginnt in einem Knoten v und endet in einem Knoten z (Bild 2.26).

Der Graph hat auch einige äußere Zweige a, b , die die bekannten Eingangsströme durch die Pins beschreiben. Eine Komponente mit p Pins hat p äußere Zweige. Ein Knoten wird als Bezugsknoten mit dem Bezugspotential festgelegt. Anstelle einer Graphischen Darstellung kann man einen solchen Graph auch durch eine Inzidenzmatrix beschreiben.

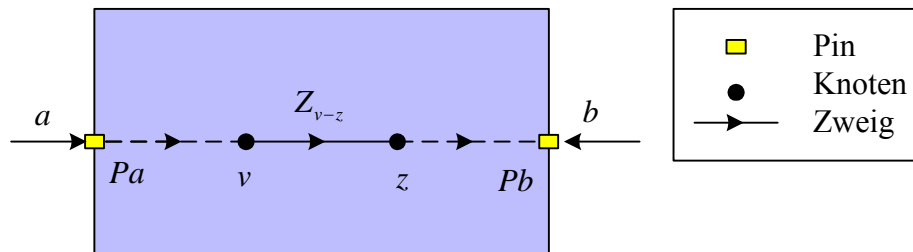


Bild 2.26: Graph des Netzwerks

Die Inzidenzmatrix zu einem gerichteten Graphen mit m Knoten und n Zweigen ist eine Matrix \mathbf{A}_W mit:

$A_w(i, j) = +1$, wenn Zweig j in Knoten i beginnt;

$A_w(i, j) = -1$, wenn Zweig j in Knoten i endet,

$A_w(i, j) = 0$, wenn Zweig j Knoten i nicht berührt.

Der Bezugsknoten wird aus der Aufstellung ausgeschlossen. Die Matrix \mathbf{A}_W hat die Dimension $n \times (m-1)$.

Diese Regel passt auch für eine Knoteninzidenzmatrix \mathbf{A}_P , die äußere Zweige beschreibt. Die Matrix \mathbf{A}_P ist eine $p \times (m-1)$ Matrix.

Die Strombilanzen nach der Kirchhoffschen Knotenregel kann man mit Hilfe der Inzidenzmatrizen \mathbf{A}_P und \mathbf{A}_W , des Vektors \mathbf{I}_Z der unbekanntenen Zweigströme und des Vektors \mathbf{I}_P der bekannten Eingangsströme offensichtlich auch in der Form

$$\mathbf{A}_P \cdot \mathbf{I}_P + \mathbf{A}_W \cdot \mathbf{I}_Z = \mathbf{0} \quad (2.62)$$

formulieren.

Beispiel 2.3

Das betrachtete R, L – Netzwerk ist in Bild 2.27 dargestellt.

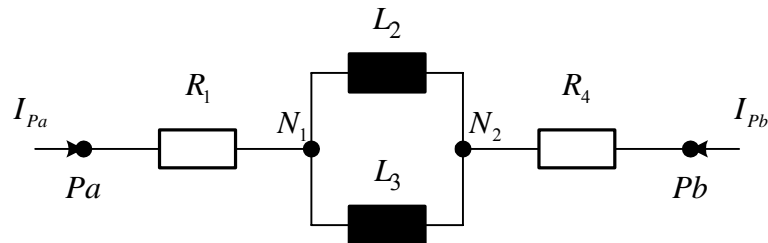


Bild 2.27: Beispiel eines R, L – Netzwerkes

Der Graph dieses Netzwerkes (Bild 2.28) besteht aus:

Pins Pa, Pb ;

Knoten $m = 4$ (Pa, Pb, N_1, N_2);

innere Zweige $n = 4$ (R_1, L_2, L_3, R_4);

äußere Zweige $p = 2$ (a, b);

Bezugsknoten N_1 .

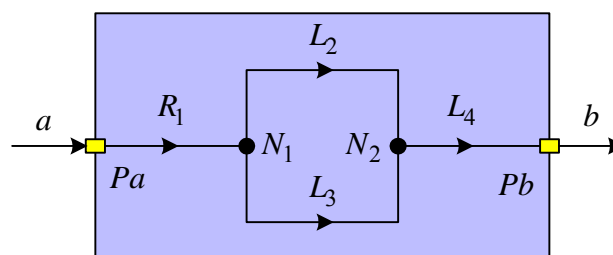


Bild 2.28: Graph des R, L -Netzwerkes

Die Inzidenzmatrizen \mathbf{A}_P und \mathbf{A}_W des gerichteten Widerstandsnetzes werden in Tabelle 2.10 gegeben. \mathbf{A}_P hat 2×3 Elemente, \mathbf{A}_W hat 4×3 Elemente.

Tabelle 2.10: Die Inzidenzmatrizen \mathbf{A}_P und \mathbf{A}_W

		Zweige					
		a	b	R_1	L_2	L_3	L_4
Knoten	Pa	-1	0	1	0	0	0
	Pb	0	-1	0	0	0	-1
	N_2	0	0	0	-1	-1	1

Man stellt aus der Tabelle 2.10 \mathbf{A}_P und \mathbf{A}_W auf:

$$\mathbf{A}_P = \begin{pmatrix} -1 & 0 \\ 0 & -1 \\ 0 & 0 \end{pmatrix}, \quad \mathbf{A}_W = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & -1 & -1 & 1 \end{pmatrix} \quad (2.63)$$

Für das Beispiel aus (2.62) ergibt dies:

$$\begin{pmatrix} -1 & 0 \\ 0 & -1 \\ 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} I_{Pa} \\ I_{Pb} \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & -1 & -1 & 1 \end{pmatrix} \cdot \begin{pmatrix} I_{R1} \\ I_{L2} \\ I_{L3} \\ I_{R4} \end{pmatrix} = 0, \quad (2.64)$$

oder in der Vektorschreibweise:

$$\begin{matrix} -I_{Pa} & & +I_{R1} \\ & -I_{Pb} & & -I_{R4} \\ & & -I_{L2} & -I_{L3} & +I_{R4} \end{matrix} = 0. \quad (2.65)$$

2.4.2 Formulierung der Spannungsgleichungen

Die Strom-Spannungsbeziehungen aller Zweige kann man in diesem Beispiel gemeinsam durch:

$$\mathbf{D}_Z \cdot \mathbf{I}_Z - \mathbf{U}_O = \mathbf{U}_Z \quad (2.66)$$

beschreiben, wobei \mathbf{I}_Z der Vektor der unbekanntten Zweigströme, \mathbf{U}_Z der Vektor der Zweigspannungen und \mathbf{D}_Z eine Diagonalmatrix mit den jeweiligen Widerstandswerten Z des Zweiges bzw. 0 im Falle der unabhängigen Spannungsquelle ist. \mathbf{U}_O ist der Vektor der unabhängigen Spannungsquellen. Die Gleichung (2.66) wird für das Beispiel 2.3 erweitert:

$$\begin{pmatrix} R_1 & 0 & 0 & 0 \\ 0 & L_2 & 0 & 0 \\ 0 & 0 & L_3 & 0 \\ 0 & 0 & 0 & R_4 \end{pmatrix} \cdot \begin{pmatrix} I_{R1} \\ I_{L2} \\ I_{L3} \\ I_{R4} \end{pmatrix} - \begin{pmatrix} 0 \\ L_2 I_{Z2}(0) \\ L_3 I_{Z3}(0) \\ 0 \end{pmatrix} = \begin{pmatrix} U_{R1} \\ U_{L2} \\ U_{L3} \\ U_{R4} \end{pmatrix}. \quad (2.67)$$

Der Vektor \mathbf{U}_O schließt auch die äquivalenten Spannungen $L_Z I_Z(0)$ ein, die den Anfangsstrom der Induktivität (2.24) beschreiben.

Aus (2.67) ergibt sich in der Vektorschreibweise:

$$\begin{array}{rcc} +R_1 \cdot I_{R1} & & 0 & U_{R1} \\ & +L_2 s \cdot I_{L2} & & -L_2 \cdot I_{L2}(0) & U_{L2} \\ & & +L_3 s \cdot I_{L3} & & -L_3 \cdot I_{L3}(0) & U_{L3} \\ & & & +R_4 \cdot I_{R4} & 0 & U_{R4} \end{array} = \begin{pmatrix} U_{R1} \\ U_{L2} \\ U_{L3} \\ U_{R4} \end{pmatrix}. \quad (2.68)$$

Die Zweigspannungen U_Z kann man auf die Knotenpotentiale P_Z zurückführen. Offensichtlich gilt:

$$\mathbf{A}_{REF} \cdot U_{REF} + \mathbf{A}_K \cdot \mathbf{P}_Z = \mathbf{U}_Z. \quad (2.69)$$

Dabei ist \mathbf{P}_Z der Vektor der Knotenpotentiale ohne das Bezugspotential, \mathbf{A}_K die Inzidenzmatrix der Knoten, U_{REF} das Bezugspotential und \mathbf{A}_{REF} der Inzidenzvektor des Bezugsknotens.

Matrix \mathbf{A}_K ist eine $(m-1) \times n$ Matrix mit:

$$A_k(i, j) = +1, \text{ wenn Zweig } j \text{ in Knoten } i \text{ beginnt;}$$

$$A_k(i, j) = -1, \text{ wenn Zweig } j \text{ in Knoten } i \text{ endet,}$$

$A_k(i, j) = 0$, wenn Zweig j Knoten i nicht berührt.

Diese Regeln gelten auch für \mathbf{A}_{REF} .

Die Inzidenzmatrix $\mathbf{A}_{\mathbf{K}}$ und der Inzidenzvektor \mathbf{A}_{REF} der Länge n für das Netzwerk aus dem Beispiel 2.3 werden in Tabelle 2.11 gegeben:

Tabelle 2.11: Die Inzidenzmatrix $\mathbf{A}_{\mathbf{K}}$ und der Inzidenzvektor \mathbf{A}_{REF} aus dem Beispiel 2.3

		Knoten			
		Pa	Pb	N_2	N_1
Zweigspannungen	U_{R1}	1	0	0	-1
	U_{L2}	0	0	-1	1
	U_{L3}	0	0	-1	1
	U_{R4}	0	-1	1	0

Im Beispiel 2.3 ergibt sich aus (2.69) also:

$$\begin{pmatrix} -1 \\ 1 \\ 1 \\ 0 \end{pmatrix} \cdot U_{N1} + \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ -1 & 0 & 0 \\ 1 & 0 & -1 \end{pmatrix} \cdot \begin{pmatrix} U_{N2} \\ U_{Pa} \\ U_{Pb} \end{pmatrix} = \begin{pmatrix} U_{R1} \\ U_{L2} \\ U_{L3} \\ U_{R4} \end{pmatrix}, \quad (2.70)$$

oder

$$\begin{array}{rcc} -U_{N1} & +U_{Pa} & U_{R1} \\ +U_{N1} & -U_{N2} & U_{L2} \\ +U_{N1} & -U_{N2} & U_{L3} \\ & +U_{N2} & -U_{Pb} & U_{R4} \end{array} = \begin{array}{c} \\ \\ \\ \end{array}. \quad (2.71)$$

2.4.3 Sparse-Tableau Formulierung der Netzwerkgleichungen

Die drei Gleichungssysteme (2.61), (2.66) und (2.69) enthalten alle Informationen um das Netzwerk zu beschreiben. Die Zusammenfassung der drei Systeme ist die Sparse-Tableau

Formulierung der Netzwerkgleichungen. Die Spannungsgleichungen (2.69) werden in die Strom-Spannungsbeziehungen (2.66) eingesetzt und damit die unbekanntenen Zweigspannungen eliminiert. Das Resultat ist bei einem Netzwerk mit m Knoten und n Zweigen ein Gleichungssystem mit $(m+n-1)$ Gleichungen und $(m+n-1)$ Unbekannten, das als Reduziertes Sparse-Tableau bezeichnet wird:

$$\begin{aligned} \mathbf{A}_P \cdot \mathbf{I}_P + \mathbf{A}_W \cdot \mathbf{I}_Z &= \mathbf{0}, \\ \mathbf{D} \cdot \mathbf{I}_Z - \mathbf{U}_O &= \mathbf{A}_{REF} \cdot U_{REF} + \mathbf{A}_K \cdot \mathbf{P}_Z. \end{aligned} \quad (2.72)$$

Man erhält aus (2.72) ein System mit den Zweigströmen \mathbf{I}_Z und Knotenpotentialen \mathbf{P}_Z als Unbekannte (Bild 2.29).

Dabei entstehen aus den elementaren Matrizen und Vektoren die größeren Matrizen \mathbf{A}_{PE} , \mathbf{T} und Vektoren U_{REF} , \mathbf{X} , U_{OE} .

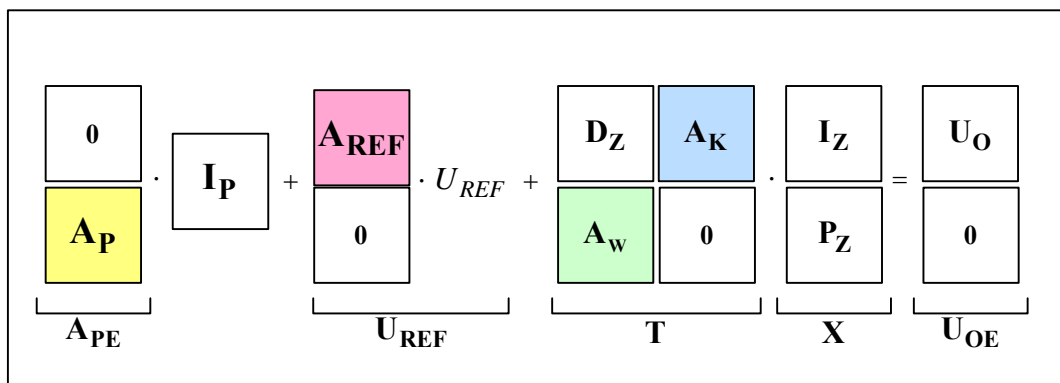


Bild 2.29: Sparse-Tableau Darstellung der Netzwerkgleichungen

Im Falle des Beispiels resultiert daraus das Gleichungssystem:

$$\begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -1 & 0 \\ 0 & -1 \\ 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} I_{Pa} \\ I_{Pb} \end{pmatrix} + \begin{pmatrix} U_{N1} \\ -U_{N1} \\ -U_{N1} \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \quad (2.73)$$

$$+ \begin{pmatrix} R_1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & L_2 s & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & L_3 s & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & R_4 & -1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & -1 & -1 & 1 & 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} I_{R1} \\ I_{L2} \\ I_{L3} \\ I_{R4} \\ U_{N2} \\ U_{Pa} \\ U_{Pb} \end{pmatrix} = \begin{pmatrix} 0 \\ L_2 \cdot I_{L2}(0) \\ L_3 \cdot I_{L3}(0) \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

Aus der Gleichung (2.73) ergeben sich $(m+n-1)$ unabhängigen Gleichungen:

$$\begin{pmatrix} R_1 \cdot I_{R1} + U_{N1} - U_{Pa} \\ L_2 s \cdot I_{L2} - U_{N1} + U_{N2} \\ L_3 s \cdot I_{L3} - U_{N1} + U_{N2} \\ R_4 \cdot I_{R4} - U_{N2} + U_{Pb} \\ I_{R1} - I_{Pa} \\ -I_{R4} - I_{Pb} \\ -I_{L2} - I_{L3} + I_{R4} \end{pmatrix} = \begin{pmatrix} 0 \\ L_2 \cdot I_{L2}(0) \\ L_3 \cdot I_{L3}(0) \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \quad (2.74)$$

2.4.4 Lösung der Tableau-Matrizengleichung

Die Tableau-Matrizen Gleichung aus dem Bild 2.29 kann man kurz umformen als

$$\mathbf{A}_{PE} \cdot \mathbf{I}_P + \mathbf{U}_{REF} + \mathbf{T} \cdot \mathbf{X} = \mathbf{U}_{OE}. \quad (2.75)$$

Dabei ist \mathbf{X} der Vektor der unbekanntem Ströme und Knotenpotentiale.

Die Matrizen Gleichung (2.75) muss nach \mathbf{X} auflöst werden. Die unbekanntem Knotenpotentiale (der Subvektor \mathbf{P}_Z im Vektor \mathbf{X}) und die Potentiale auf den Pins kann man durch den Vektor der Eingangsströme \mathbf{I}_P und das Referenzpotential U_{REF} ausdrücken:

$$\mathbf{P}_Z = f(\mathbf{I}_P, U_{REF}) \quad (2.76)$$

Ein Vergleich von (2.61) mit der Gleichung in der Form (2.76) ergibt die Parameter U^* , R^* , L^* für die analysierte Netzwerkstruktur der Komponente.

Umgeformt ergibt sich aus (2.75):

$$\mathbf{T} \cdot \mathbf{X} = \mathbf{U}_{\text{OE}} - \mathbf{A}_{\text{PE}} \cdot \mathbf{I}_{\text{P}} - \mathbf{U}_{\text{REF}}. \quad (2.77)$$

Das System ist lösbar, falls die Determinante von \mathbf{T} nicht gleich Null ist. Falls die Determinante von \mathbf{T} gleich Null ist, wurde die Matrizen falsch aufgestellt.

Nach Anwendung der Gauß-Elimination auf die Gleichung (2.77) erhält man

$$\mathbf{T}_{\text{U}} \cdot \mathbf{X} = \mathbf{B}. \quad (2.78)$$

Dabei ist \mathbf{T}_{U} eine rechte obere Dreiecksmatrix, also eine quadratische Matrix, für die alle Einträge unterhalb der Diagonalelemente gleich Null sind.

Die Auflösung der Matrixgleichung (2.78) nach \mathbf{X} ergibt:

$$\mathbf{X} = \tilde{\mathbf{B}}. \quad (2.79)$$

Die Gleichungen (2.78) und (2.79) bilden den Algorithmus *GaussCalc*.

Anwendung der Gauß-Elimination auf die Gleichungen (2.73) (Beispiel 2.3) ergibt sich für die Matrixgleichung (2.78):

$$\begin{pmatrix} R_1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & L_2 s & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & L_3 s & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & R_4 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & -\frac{1}{R_4} & 0 & \frac{1}{R_4} \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{R_1} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{L_2 + L_3}{L_2 L_3} s \end{pmatrix} \cdot \begin{pmatrix} I_{R1} \\ I_{L2} \\ I_{L3} \\ I_{R4} \\ U_{N2} \\ U_{Pa} \\ U_{Pb} \end{pmatrix} = \begin{pmatrix} -U_{N1} \\ L_2 I_{L2}(0) + U_{N1} \\ L_3 I_{L3}(0) + U_{N1} \\ 0 \\ I_{Pb} \\ I_{Pa} + \frac{U_{N1}}{R_1} \\ B_{7,1} \end{pmatrix}, \quad (2.80)$$

$$B_{7,1} = \left(1 + R_4 \frac{L_2 + L_3}{L_2 L_3} s\right) I_{Pb} + \left(\frac{L_2 + L_3}{L_2 L_3} s\right) U_{N1} + (I_{L2}(0) + I_{L3}(0)) \frac{L_2 + L_3}{L_2 L_3} s.$$

und danach:

$$\begin{aligned} I_{R1} &= \frac{-U_{N1} + U_{Pa}}{R_1}, \\ I_{L2} &= \frac{U_{N1} + L_2 I_{L2}(0) - U_{N2}}{L_2 s}, \\ I_{L3} &= \frac{U_{N1} + L_3 I_{L3}(0) - U_{N2}}{L_3 s}, \\ I_{R4} &= \frac{U_{N2} - U_{Pb}}{R_4}, \\ U_{N2} &= -R_4 I_{Pb} + U_{Pb}, \\ U_{Pa} &= R_1 I_{Pa} + U_{N1}, \\ U_{Pb} &= \left(\frac{L_2 L_3}{L_2 + L_3} s + R_4\right) I_{Pb} + U_{N1} + \frac{L_2 L_3}{L_2 + L_3} (I_{L2}(0) + I_{L3}(0)). \end{aligned} \tag{2.81}$$

2.4.5 Ermittlung der Schnittstellenparameter

Die Übertragungsfunktion $G(s)$ des komplexen Netzwerkes mit k geschlossenen Maschen ist eine rationale Funktion [FOL94]:

$$G(s) = \frac{\rho_0 + \rho_1 s + \dots + \rho_k s^k}{\eta_0 + \eta_1 s + \dots + \eta_{k-1} s^{k-1}}. \tag{2.82}$$

Dabei sei $\eta_{k-1} \neq 0$ und mindestens ein $\rho_v \neq 0$. Zähler und Nenner seien stets ohne gemeinsame Nullstelle.

Aus der Lösung (2.79) ergibt sich der Vektor der Knotenpotentiale \mathbf{P}_Z , er enthält beliebige Schnittstellenspannungen U_ν am Pin ν , im Bildbereich erhält man:

$$U_v(s) = U_{REF}(s) + G_{va}(s) \cdot I_a(s) + G_{vb}(s) \cdot I_b(s) \dots + G_{vp}(s) \cdot I_p(s),$$

$$U_v(s) = U_{REF}(s) + \sum_{j \in N} G_{vj}(s) \cdot I_j(s) = U_{REF}(s) + \frac{P_v(s)}{Q_v(s)}, \quad (2.83)$$

$$\text{wobei } N = \{a, b, \dots, p\}.$$

Der rationale Bruch $\frac{P_v(s)}{Q_v(s)}$ hängt von den Schnittstellenströmen ab:

$$\frac{P_v(s)}{Q_v(s)} = f(I_a, I_b, \dots, I_p, s),$$

$$P_v(s) = \sum_{\sigma=0}^{\lambda_v+1} p_{v\sigma} s^\sigma, \quad Q_v(s) = \sum_{\sigma=0}^{\lambda_i} q_{v\sigma} s^\sigma, \quad (2.84)$$

$$\lambda_v \leq k-1.$$

Dabei ist λ_v der Maximale Grad des Polynoms für die Spannung U_v an Pin v . Der Leitkoeffizient $q_{v\lambda_v} \neq 0$. Mindestens ein Koeffizient $p_{v\sigma}$ des Polynoms $P_v(s)$ muss ungleich null sein.

In den folgenden Formeln wird (2.83) und (2.84) ohne Indizes formuliert, um die Ausdrücke kurz und deutlich darzustellen. Es wird aber gemeint, dass man die Spannungsgleichung für den Pin v betrachtet:

$$U(s) = U_{REF}(s) + \frac{P(s)}{Q(s)}, \quad (2.85)$$

$$P(s) = \sum_{\sigma=0}^{\lambda+1} p_\sigma s^\sigma, \quad Q(s) = \sum_{\sigma=0}^{\lambda} q_\sigma s^\sigma, \quad \lambda \leq k-1.$$

Die Parameter U^* , R^* , L^* können aus dem Vergleich von (2.85) mit (2.33) gefunden werden.

Man wendet die Polynomdivision auf den Bruch $\frac{P(s)}{Q(s)}$ an:

$$\frac{P(s)}{Q(s)} = A(s) + \frac{B(s)}{Q(s)}, \quad (2.86)$$

$$U(s) = U_{REF} + A(s) + \frac{B(s)}{Q(s)},$$

$$A(s) = a_0 + a_1 s, \quad B(s) = \sum_{\sigma=0}^{\lambda-1} b_{\sigma} s^{\sigma}.$$

Dabei ist $A(s)$ der Quotient und $B(s)$ der Rest. Die Koeffizienten $b_0 \dots b_{\lambda-1}$ können gleich Null sein.

Die Koeffizienten des Polynoms $A(s)$ hängen von den Eingangsströmen ab

$$\begin{aligned} a_0 &= \tilde{a}_{00} + \tilde{a}_{0a} I_a(s) + \tilde{a}_{0b} I_b(s) + \dots + \tilde{a}_{0p} I_p(s), \\ a_1 &= \tilde{a}_{1a} I_a(s) + \tilde{a}_{1b} I_b(s) + \dots + \tilde{a}_{1p} I_p(s). \end{aligned} \quad (2.87)$$

Beim Vergleich von $A(s)$ mit (2.33) ergibt sich die exakte Lösung für die Parameter R^* und L^* . In der Gleichung (2.87) sind die Vektoren \mathbf{R}^* und \mathbf{L}^* die Koeffizienten des Ausdruckes $I(s)$ und $s I(s)$:

$$\mathbf{R}^* = \begin{pmatrix} \tilde{a}_{0a} \\ \tilde{a}_{0p} \\ \vdots \\ \tilde{a}_{0p} \end{pmatrix}, \quad \mathbf{L}^* = \begin{pmatrix} \tilde{a}_{1a} \\ \tilde{a}_{1p} \\ \vdots \\ \tilde{a}_{1p} \end{pmatrix}. \quad (2.88)$$

Die Gleichung (2.86) enthält noch den Teil $\frac{B(s)}{Q(s)}$, den man als $Y(s)$ bezeichnet:

$$Y(s) = \frac{B(s)}{Q(s)}. \quad (2.89)$$

Nach dem Übergang in den Zeitbereich und der Ermittlung, kann Y dann in den Parameter U^* eingesetzt werden. U^* besteht aus zwei Teilen:

$$U^* = \tilde{a}_{00} + Y. \quad (2.90)$$

Der Zählergrad der komplexen rationalen Funktion $Y(s)$ ist kleiner als der Nennergrad.

Weiterhin folgt aus (2.89):

$$Q(s) \cdot Y(s) = B(s),$$

$$\left(\sum_{\sigma=0}^{\lambda} q_{\sigma} s^{\sigma} \right) Y(s) = \sum_{\sigma=0}^{\lambda-1} b_{\sigma} s^{\sigma}. \quad (2.91)$$

$Y(s)$ ist vom Strom $I(s)$ abhängig. Im Zeitbereich müssen die Variable Y und ihre Ableitungen $\dot{Y} \dots Y^{(k)}$ berechnet werden.

Die Zustandsgleichung (2.91) wird in die Beobachtungsnormalform umgesetzt [FOL94].

Man nimmt zusätzlich an, dass der Leitkoeffizient im Polynom $Q(s)$ $q_{\lambda} = 1$ sei.

Die Gleichung (2.91) wird durch $q_{\lambda} s^{\lambda}$ dividiert:

$$\left(\frac{q_0}{q_{\lambda}} \frac{1}{s^{\lambda}} + \frac{q_1}{q_{\lambda}} \frac{1}{s^{\lambda-1}} + \dots + \frac{q_{\lambda-1}}{q_{\lambda}} \frac{1}{s} + 1 \right) Y(s) = \left(\frac{b_0}{q_{\lambda}} \frac{1}{s^{\lambda}} + \frac{b_1}{q_{\lambda}} \frac{1}{s^{\lambda-1}} + \dots + \frac{b_{\lambda-1}}{q_{\lambda}} \frac{1}{s} \right) \cdot I(s) \quad (2.92)$$

oder

$$Y(s) = \frac{1}{s} \left\{ \frac{b_{\lambda-1}}{q_{\lambda}} I(s) - \frac{q_{\lambda-1}}{q_{\lambda}} Y(s) + \right.$$

$$\left. + \frac{1}{s} \left\langle \dots + \frac{1}{s} \left[\frac{b_1}{q_{\lambda}} I(s) - \frac{q_1}{q_{\lambda}} Y(s) + \frac{1}{s} \left(\frac{b_0}{q_{\lambda}} I(s) - \frac{q_0}{q_{\lambda}} Y(s) \right) \right] \dots \right\rangle \right\}. \quad (2.93)$$

Nunmehr definiert man als Zustandsvariablen die mit $1/s$ multiplizierten Klammerausdrücke, wobei man mit der innersten Klammer beginnt und $Y(s) = H_{\lambda}(s)$ setzt:

$$H_1(s) = \frac{1}{s} \left[\frac{b_0}{q_{\lambda}} \cdot I(s) - \frac{q_0}{q_{\lambda}} \cdot H_{\lambda}(s) \right],$$

$$H_2(s) = \frac{1}{s} \left[\frac{b_1}{q_{\lambda}} \cdot I(s) - \frac{q_1}{q_{\lambda}} \cdot H_{\lambda}(s) + H_1(s) \right], \quad (2.94)$$

$$H_\lambda(s) = \frac{1}{s} \left[\begin{array}{c} \vdots \\ \frac{b_{\lambda-1}}{q_\lambda} \cdot I(s) - \frac{q_{\lambda-1}}{q_\lambda} \cdot H_\lambda(s) + H_{\lambda-1}(s) \end{array} \right].$$

Aus der Multiplikation mit s und dem Übergang in den Zeitbereich folgt:

$$\begin{aligned} \dot{H}_1 &= -\frac{q_0}{q_\lambda} \cdot H_\lambda + \frac{b_0}{q_\lambda} \cdot I, \\ \dot{H}_2 &= -\frac{q_1}{q_\lambda} \cdot H_\lambda + H_1 + \frac{b_1}{q_\lambda} \cdot I, \\ &\vdots \\ \dot{H}_k &= -\frac{q_{\lambda-1}}{q_\lambda} \cdot H_\lambda + H_{\lambda-1} + \frac{b_{\lambda-1}}{q_\lambda} \cdot I. \end{aligned} \quad (2.95)$$

In Matrixschreibweise wird daraus die Vektordifferentialgleichung im Zeitbereich:

$$\dot{\mathbf{H}} = \begin{pmatrix} 0 & 0 & \dots & -\frac{q_0}{q_\lambda} \\ 1 & 0 & & -\frac{q_1}{q_\lambda} \\ & 1 & & -\frac{q_2}{q_\lambda} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 - \frac{q_{\lambda-1}}{q_\lambda} \end{pmatrix} \cdot \mathbf{H} + \begin{pmatrix} \frac{b_0}{q_\lambda} \\ \frac{b_1}{q_\lambda} \\ \frac{b_2}{q_\lambda} \\ \vdots \\ \frac{b_{\lambda-1}}{q_\lambda} \end{pmatrix} \cdot I. \quad (2.96)$$

Die Ausgangsgleichung ist $Y = H_\lambda$ oder in Vektorform:

$$Y = (0 \quad \dots \quad 0 \quad 1) \cdot \mathbf{H}. \quad (2.97)$$

Die Zustandsgleichungen (2.96) und (2.97) bezeichnet man als Beobachtungsnormalform.

Die Lösung des Systems (2.96) ergibt den Vektor \mathbf{H} und danach Y , dies ergänzt den Parameter U^* .

Man ermittelt die Schnittstellenparameter aus der Lösung des Systems der Gleichungen für das Beispiel 2.3.

Die Matrizen R^* , L^* und den Vektor U^* der Parameter aus der Gleichung (2.81) für U_{Pa} :

$$\begin{cases} U_{Pa} = U_{REF} + U_{Pa}^* + (R_{Paa}^* + L_{Paa}^* s) I_{Pa}, \\ U_{Pa} = U_{N1} + (R_1) I_{Pa}; \end{cases} \quad (2.98)$$

$$\Downarrow$$

$$U_{Pa}^* = 0, R_{Paa}^* = R_1, L_{Paa}^* = 0.$$

und für U_{Pb} :

$$\begin{cases} U_{Pb} = U_{REF} + U_{Pb}^* + (R_{Pbb}^* + L_{Pbb}^* s) I_{Pb}, \\ U_{Pb} = U_{N1} + \frac{L_2 L_3}{L_2 + L_3} (I_{L2}(0) + I_{L3}(0)) + \left(R_4 + \frac{L_2 L_3}{L_2 + L_3} s \right) I_{Pb}, \end{cases} \quad (2.99)$$

$$\Downarrow$$

$$U_{Pb}^* = \frac{L_2 L_3}{L_2 + L_3} (I_{L2}(0) + I_{L3}(0)), R_{Pbb}^* = R_4, L_{Pbb}^* = \frac{L_2 L_3}{L_2 + L_3}.$$

Die errechneten Parameter werden in Tabelle 2.12 dargestellt:

Tabelle 2.12 – Errechnete Parameter der Komponente des Beispiels 2.3

Schnittstellenparameter	Werte
U^*	$\begin{pmatrix} 0 & 0 \\ 0 & \frac{L_2 L_3}{L_2 + L_3} (I_{L2}(0) + I_{L3}(0)) \end{pmatrix}$
R^*	$\begin{pmatrix} R_1 & 0 \\ 0 & R_4 \end{pmatrix}$
L^*	$\begin{pmatrix} 0 & 0 \\ 0 & \frac{L_2 L_3}{L_2 + L_3} \end{pmatrix}$

Durch die Analyse der erhaltenen Parameter lässt sich prüfen, ob diese Parameter physikalisch richtig sind. Das äquivalente Ersetzungsschema des Netzwerkes mit Hilfe der aufgelösten Parameter besteht aus zwei Ersatzwiderständen R_{Paa}^* , R_{Pbb}^* und einer Ersatzinduktivität L_{Pbb}^* , die aus zwei parallel geschalteten Spulen besteht (Bild 2.30). Der Anfangsstrom in der Ersatzinduktivität L_{Pbb}^* ist $I_{L2}(0) + I_{L3}(0)$.

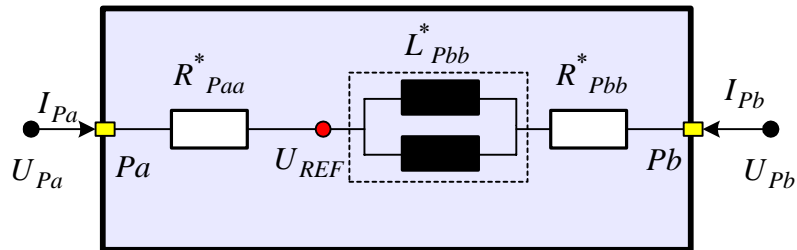


Bild 2.30 – Äquivalente Schaltung des Beispielkreises 2.3

Ein Vergleich der gefundenen Parameterwerte und des Ersetzungsschemas mit dem Anfangsschema in Bild 2.27 zeigt die Richtigkeit der Reihenfolge der Parameterberechnung.

2.4.6 Anwendung der Parameterberechnungsmethodik bei der Strukturkomponente

Die betrachtete Reihenfolge der Parameterberechnung ist auch für die Netzwerke mit den inneren Strukturkomponenten gültig. In Kapitel 2.1 wird erwähnt, dass eine Strukturkomponente aus der Menge der Grundkomponenten und Strukturkomponenten der niedrigeren Ebene bestehen kann.

Ein Strukturblock SUB mit p Pins kann durch die p Ersatzgrundkomponente und einen Referenzknoten des Strukturblocks U_{REFSUB} ersetzt werden (Bild 2.31). Die Ersatzgrundkomponente beschreibt die äquivalente Schaltung vom Referenzknoten U_{REFSUB} bis zum Pin $SUBv$. Die Ersatzgrundkomponente wird durch die Parameter U_{SUBv}^* , \mathbf{R}_{SUBv}^* , \mathbf{L}_{SUBv}^* der Strukturkomponente bestimmt. U_{SUBv}^* ist das Element v des Vektors des Parameters \mathbf{U}^* . Der Vektor \mathbf{R}_{SUBv}^* ist die Spalte v der Matrix des Parameters \mathbf{R}^* . Der Vektor \mathbf{L}_{SUBv}^* ist die Spalte v der Matrix des Parameters \mathbf{L}^* .

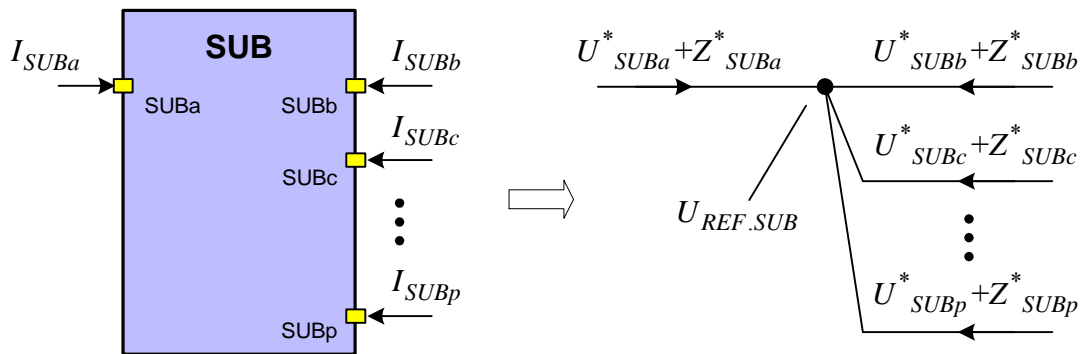


Bild 2.31 – Strukturkomponente und der äquivalente Graph

Die inneren Strukturkomponenten werden mit folgenden Annahmen berücksichtigt.

Die Menge der Knoten des Netzwerks m besteht dann aus drei Gruppen der Knoten:

- Knoten, die die Komponente des Netzwerks verbinden,
- Knoten, die den Pins entsprechen,
- Referenzknoten der inneren Strukturkomponenten.

Die Potentiale der Referenzknoten der inneren Strukturkomponenten werden in den Vektor \mathbf{P}_Z hinzugefügt.

Die Eingangsströme der inneren Strukturkomponenten werden in den Vektor \mathbf{I}_Z hinzugefügt.

Die Matrix \mathbf{D}_Z und der Vektor \mathbf{U}_O werden mit den Elementen U^*_{SUBv} , R^*_{SUBvv} , L^*_{SUBvv} der Zweige der äquivalenten Ersatzschaltung des Blocks ausgefüllt. Die Elemente R^*_{SUBvv} , L^*_{SUBvv} werden sowohl in der Hauptdiagonale der Matrix \mathbf{D}_Z , stehen als auch an den Stellen, die den Eingangsströmen der Strukturkomponente entsprechen.

Die Stromgleichungen für die Referenzknoten des Strukturblocks werden ähnlich wie für die anderen Knoten formuliert.

2.4.7 Kurze Zusammenfassung der Methodik der Parameterermittlung

Die Schritte der Methodik der Parameterermittlung, die in den Abschnitten 2.4.1-2.4.5 geschrieben wurde, werden zusammengefasst [YEF05] und in Bild 2.32 dargestellt.

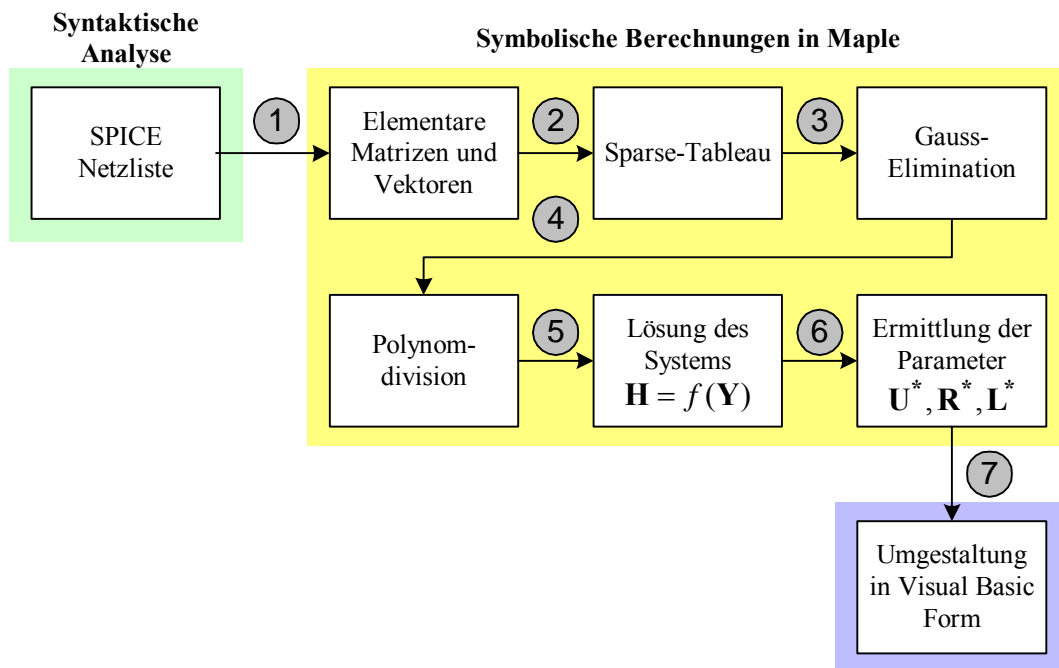


Bild 2.32: Blockschema der Ermittlung der Schnittstellenparameter

1. Aus SPICE ähnlicher Netzliste oder XML – basierter Netzliste auf der Basis der Topologie des Netzwerkes werden die elementaren Matrizen \mathbf{A}_p , \mathbf{A}_w , \mathbf{D} , \mathbf{A}_K und die Vektoren \mathbf{I}_Z , \mathbf{I}_p , \mathbf{P}_Z , \mathbf{U}_O , \mathbf{A}_{REF} aufgestellt.
2. Aus den elementaren Matrizen und Vektoren wird das Sparse-Tableau mit den Matrizen \mathbf{A}_{PE} , \mathbf{T} und Vektoren \mathbf{U}_{REF} , \mathbf{X} , \mathbf{U}_{OE} aufgebaut.
3. Anwendung der Gauß-Elimination und Auflösung der Matrixgleichungen (2.77), (2.78) in symbolischer Form ergibt die Gleichungen, die die gesuchten Parameter enthalten.
4. Die Polynomdivision ergibt die Vektoren des Quotienten $\mathbf{A}(s)$, des Rests $\mathbf{B}(s)$ und des Teilers $\mathbf{Q}(s)$.
5. Das System (2.96) wird nach Y aufgelöst.
6. Die Parameter U^* , R^* , L^* werden aus den Koeffizienten ermittelt.
7. Die erhaltenen Ausdrücke werden aus der Maple – Darstellung in die Visual Basic Syntax überführt.

Der Schritt 1 kann in Maple oder in der speziellen Anwendung gemacht werden. Die Schritte 2 – 6 werden symbolisch in Maple durchgeführt. Der Schritt 7 wird mittels einer speziellen

Funktion in Maple erledigt. Die Schritte 4 - 6 werden durch den Algorithmus *ParamExtr* bezeichnet.

2.5 Berechnung der Unbekannten

2.5.1 Reihenfolge der Berechnung

Man betrachtet ein System mit n hierarchischen Ebenen. Zwei verbundene Komponenten P und SUB bilden zwei oberste hierarchische Ebenen (Bild 2.33).

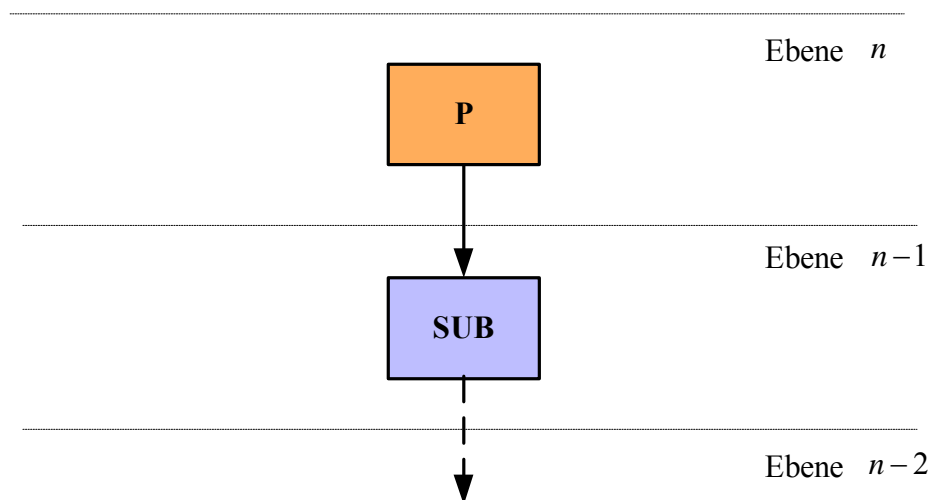


Bild 2.33: Beziehungen „Parentblock“ – „Subblock“

Die Komponente P gehört zur Ebene n und die Komponente SUB gehört zur Ebene $n-1$. Darunter liegen noch $n-2$ Ebenen mit den hierarchisch verbundenen Komponenten. Man nennt die Komponente P für die unten liegende Komponente SUB „Parentblock“. Die Komponente SUB für die oben liegende Komponente P bezeichnet man als „Subblock“. Solche Beziehungen zwischen den Komponenten gelten für jede hierarchische Ebene des Systems.

Es gibt zwei mögliche Reihenfolgen der Berechnung. Zuerst betrachtet man den Fall, dass der Subblock keine Induktivitäten enthält und deshalb die inneren Ströme des Subblocks und alle Anschlussströme im beliebigen Zeitpunkt 0 nicht bekannt sind. Diese Ströme müssen auf der oberen Ebene gerechnet werden.

Der Block SUB wird mit den Schnittstellenparametern entsprechend der Methodik der Parameterberechnung definiert (Abschnitt 2.2). Diese Parameter sind der Vektor \mathbf{U}_{SUB}^* und die Matrizen \mathbf{R}_{SUB}^* und \mathbf{L}_{SUB}^* .

Alle Elemente der Matrix \mathbf{L}_{SUB}^* sind gleich Null. Aus der Gleichung der Schnittstellendefinition (2.33) ergibt sich:

$$\mathbf{U}_{SUB} = \mathbf{U}_{REF.SUB} + \mathbf{U}_{SUB}^* + \mathbf{R}_{SUB}^* \cdot \mathbf{I}_{SUB}. \quad (2.100)$$

Dabei ist $\mathbf{U}_{REF.SUB}$ der Vektor der Referenzpotentiale des Blocks SUB . \mathbf{I}_{SUB} und \mathbf{U}_{SUB} sind die Vektoren der Eingangsströme und Eingangsspannungen des Blocks SUB entsprechend.

Der Vektor \mathbf{U}_P^* und die Matrizen \mathbf{R}_P^* , \mathbf{L}_P^* definieren die Schnittstellenparameter an den Pins des Blocks P . \mathbf{I}_P ist der Vektor der Eingangsströme des Blocks P . $\dot{\mathbf{I}}_P$ ist der Vektor der Ableitungen der Eingangsströme. \mathbf{U}_P ist der Vektor der Eingangsspannungen:

$$\mathbf{U}_P = \mathbf{U}_{REF.P} + \mathbf{U}_P^* + \mathbf{R}_P^* \cdot \mathbf{I}_P + \mathbf{L}_P^* \cdot \dot{\mathbf{I}}_P. \quad (2.101)$$

Dabei ist $\mathbf{U}_{REF.P}$ der Vektor der Referenzpotentiale des Blocks P .

Die Reihenfolge der Strom- und Spannungsberechnungen wird in fünf Schritten betrachtet [KAS05] (Bild 2.34).

Schritt 1

Der Subblock empfängt die Schnittstellenparameter von seinen Subblöcken niedrigerer Ebene und berechnet auf der Basis seiner Topologie die Schnittstellenparameter \mathbf{U}_{SUB}^* , \mathbf{R}_{SUB}^* , \mathbf{L}_{SUB}^* . Diesen Vektor und die Matrizen sendet er zu seinem Parentblock.

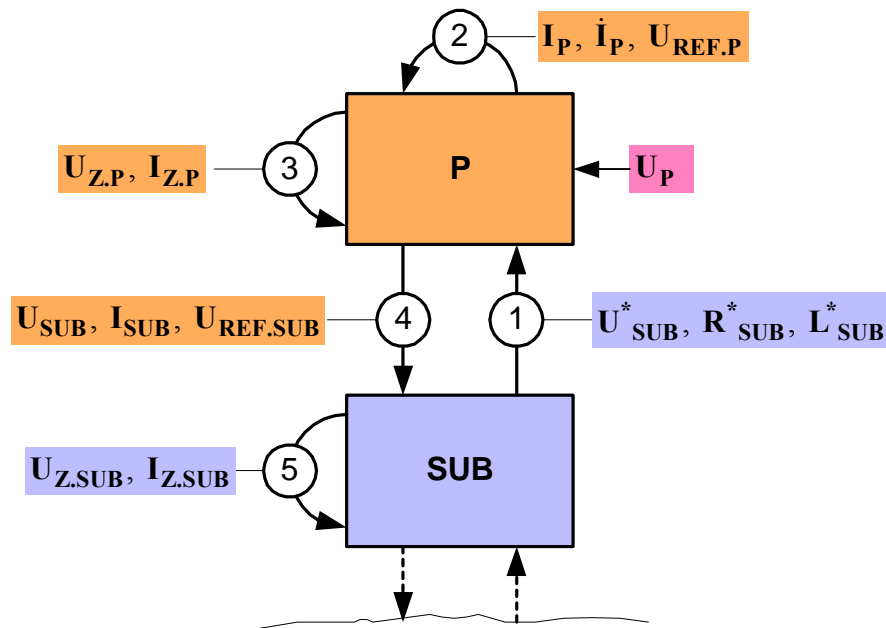


Bild 2.34: Reihenfolge der Berechnung beim Subblock ohne Induktivitäten

Schritt 2

Der Block P empfängt den Vektor und die Matrizen $\mathbf{U}_{\text{SUB}}^*$, $\mathbf{R}_{\text{SUB}}^*$, $\mathbf{L}_{\text{SUB}}^*$ aus dem Subblock und berechnet seine Parameter \mathbf{U}_{P}^* , \mathbf{R}_{P}^* , \mathbf{L}_{P}^* . Dann erhält er die Werte der Spannungen an seinen Pins (der Vektor \mathbf{U}_{P}) und berechnet die Ströme, ihre Ableitungen und das Referenzpotential (die Vektoren \mathbf{I}_{P} , $\dot{\mathbf{I}}_{\text{P}}$, $\mathbf{U}_{\text{REF.P}}$).

Schritt 3

Die inneren Ströme und Spannungen der Komponente P (die Vektoren $\mathbf{U}_{\text{Z.P}}$, $\mathbf{I}_{\text{Z.P}}$) werden berechnet. Unter diesen sind auch der Vektor der Eingangsströme \mathbf{I}_{SUB} , der Vektor der Eingangsspannungen \mathbf{U}_{SUB} und der Vektor der Referenzpotentiale $\mathbf{U}_{\text{REF.SUB}}$ des Subblocks.

Schritt 4

Der Parentblock schickt die berechneten Vektoren \mathbf{I}_{SUB} , \mathbf{U}_{SUB} , $\mathbf{U}_{\text{REF.SUB}}$ zu seinem Subblock.

Schritt 5

Der Subblock erhält die Vektoren \mathbf{I}_{SUB} , \mathbf{U}_{SUB} , $\mathbf{U}_{\text{REF.SUB}}$ und berechnet die inneren Ströme und Spannungen ($\mathbf{U}_{\text{Z.SUB}}$, $\mathbf{I}_{\text{Z.SUB}}$).

Die errechneten Vektoren der Ströme und Spannungen werden nacheinander zu den unteren Subblöcken gesendet.

Wenn der Subblock Induktivitäten enthält, dann sind die Induktivitätsströme die Zustandsgrößen. Sie sind in jeden Zeitpunkt bekannt. Der Strom an den Pins des Subblocks $\mathbf{I}_{\text{SUB}_v}$ kann durch die Induktivitätsströme des Subblocks definiert und errechnet werden, falls der Strom $\mathbf{I}_{\text{SUB}_v}$ eine Zustandsgröße ist. Im Zeitpunkt 0 sind die Anfangswerte der Induktivitätsströme bekannt (der Vektor $\mathbf{I}_{\text{L.SUB}}(0)$). Der Block *SUB* wird mit dem Vektor $\mathbf{U}_{\text{SUB}}^*$ und der Matrizen $\mathbf{R}_{\text{SUB}}^*$ und $\mathbf{L}_{\text{SUB}}^*$ definiert. Aus der Gleichung der Schnittstellendefinition (2.33) ergibt sich:

$$\mathbf{U}_{\text{SUB}} = \mathbf{U}_{\text{REF.SUB}} + \mathbf{U}_{\text{SUB}}^* + \mathbf{R}_{\text{SUB}}^* \cdot \mathbf{I}_{\text{SUB}} + \mathbf{L}_{\text{SUB}}^* \cdot \dot{\mathbf{I}}_{\text{SUB}}. \quad (2.102)$$

Dabei ist $\dot{\mathbf{I}}_{\text{SUB}}$ der Vektor der Ableitungen der Eingangsströme.

Die Reihenfolge der Berechnung wird in Bild 2.35 dargestellt.

Schritt 1

Der Subblock empfängt die Schnittstellenparameter und berechneten Anschlussströme von seinen Subblöcken niedrigerer Ebene und berechnet auf der Basis seiner Topologie die Schnittstellenparameter $\mathbf{U}_{\text{SUB}}^*$, $\mathbf{R}_{\text{SUB}}^*$, $\mathbf{L}_{\text{SUB}}^*$. Diesen Vektor und die Matrizen sendet er zu seinem Parentblock. Die Komponente errechnet die Ströme an den Pins \mathbf{I}_{SUB} (falls die die Zustandsgrößen sind) und liefert diese zu seinem Parentblock.

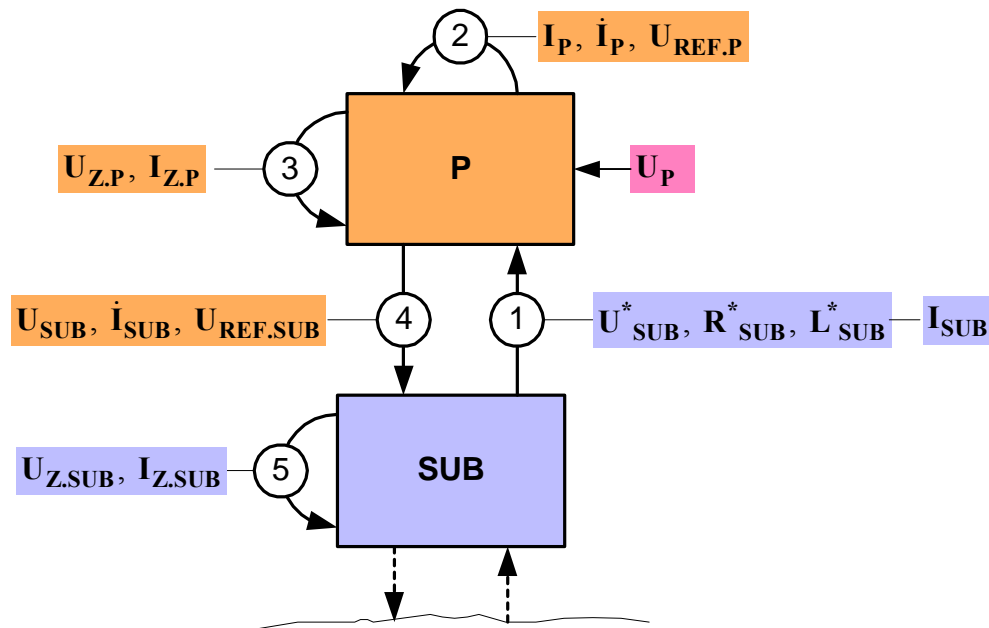


Bild 2.35: Reihenfolge der Berechnung des Subblocks mit den Induktivitäten

Schritt 2

Der Block P empfängt U_{SUB}^* , R_{SUB}^* , L_{SUB}^* , I_{SUB} aus dem Subblock und berechnet seine Parameter U_P^* , R_P^* , L_P^* . Dann erhält er die Werte der Spannungen an seinen Pins (der Vektor U_P) und berechnet die Ströme, ihre Ableitungen und das Referenzpotential (die Vektoren I_P , \dot{I}_P , $U_{\text{REF.P}}$).

Schritt 3

Die inneren Ströme und Spannungen der Komponente P (die Vektoren $U_{Z.P}$, $I_{Z.P}$) werden berechnet. Unter diesen sind \dot{I}_{SUB} , U_{SUB} , $U_{\text{REF.SUB}}$.

Schritt 4

Der Parentblock übermittelt die berechneten Vektoren \dot{I}_{SUB} , U_{SUB} , $U_{\text{REF.SUB}}$ zu seinem Subblock.

Schritt 5

Der Subblock erhält die Vektoren $\dot{\mathbf{I}}_{\text{SUB}}$, \mathbf{U}_{SUB} , $\mathbf{U}_{\text{REF.SUB}}$ und berechnet die inneren Ströme und Spannungen ($\mathbf{U}_{\text{Z.SUB}}$, $\mathbf{I}_{\text{Z.SUB}}$). Die neuen Werte des Vektors $\mathbf{I}_{\text{L.SUB}}$ werden errechnet.

Bei einer realen Komponente mit inneren Induktivitäten sind einige Anschlussspannungen nicht von den Ableitungen der Eingangsströme abhängig (einige Spalten der Matrix \mathbf{L}^* sind gleich Null). Dies hängt von der Stelle des Referenzknotens ab. Das gesamte Netzwerk wird im Referenzpunkt in mehrere separate Anteile gegliedert. Einige Anteile können nur Widerstandscharakter haben (s. der Abschnitt 2.2.2).

So ist zum Beispiel die Spannung U_b für den Kreis in Bild 2.36 ist von den Ableitungen der Ströme unabhängig:

$$U_b = U_{REF} + R_2 \cdot I_b. \quad (2.103)$$

Man nennt den Pin, bei dem mindestens ein Parameter L^* nicht gleich Null ist, „Pin mit offener Induktivität“. Das heißt, die innere Induktivität wirkt durch den Pin auf den Parentblock mittels der Ableitung des Induktivitätsstromes. Diese muss im Parentblock berücksichtigt werden.

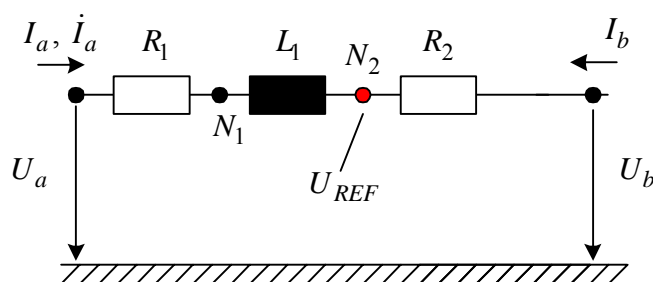


Bild 2.36: Aufteilung des Netzwerkes mit dem Referenzpunkt

Die allgemeine Reihenfolge der Stromberechnung vereinigt beide Fälle:

- Die Ströme an den Pins mit den offenen Induktivitäten werden im Subblock errechnet und nach oben zum Parentblock geliefert. Der Parentblock berechnet die Werte der Ableitungen der Anschlussströme und übermittelt diese zurück zum Subblock;
- Die Ströme an den Pins ohne offene Induktivitäten werden im Parentblock berechnet und nach unten zum Subblock gesendet.

2.5.2 Berechnung der Schnittstellenströme auf der obersten Ebene

Man betrachtet die Stromberechnung auf der obersten Ebene des modellierten Systems. Die Komponente hat p Pins, die an den Blocks mit den bekannten Spannungen U_a, U_b, \dots, U_p angeschlossen sind (Bild 2.37).

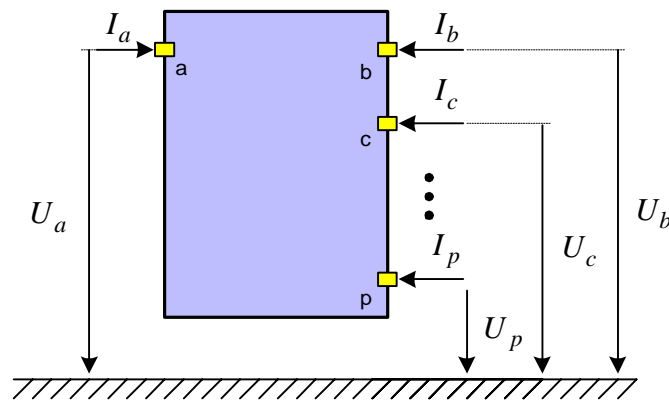


Bild 2.37: Berechnung der Schnittstellenströme auf der obersten Ebene

Die Matrixgleichung (2.33) im Zeitbereich für die Komponente mit p Pins ist:

$$\begin{pmatrix} U_a \\ U_b \\ \vdots \\ U_p \end{pmatrix} = \begin{pmatrix} U_{REF} \\ U_{REF} \\ \vdots \\ U_{REF} \end{pmatrix} + \begin{pmatrix} U_a^* \\ U_b^* \\ \vdots \\ U_p^* \end{pmatrix} + \begin{pmatrix} R_{aa}^* & R_{ab}^* & \dots & R_{ap}^* \\ R_{ba}^* & Z_{bb}^* & \dots & R_{bp}^* \\ \vdots & \vdots & \ddots & \vdots \\ R_{pa}^* & R_{pb}^* & \dots & R_{pp}^* \end{pmatrix} \cdot \begin{pmatrix} I_a \\ I_b \\ \vdots \\ I_p \end{pmatrix} + \quad (2.104)$$

$$+ \begin{pmatrix} L_{aa}^* & L_{ab}^* & \dots & L_{ap}^* \\ L_{ba}^* & L_{bb}^* & \dots & L_{bp}^* \\ \vdots & \vdots & \ddots & \vdots \\ L_{pa}^* & L_{pb}^* & \dots & L_{pp}^* \end{pmatrix} \cdot \begin{pmatrix} \dot{i}_a \\ \dot{i}_b \\ \vdots \\ \dot{i}_p \end{pmatrix}.$$

Für die Ströme aus (2.34) gilt:

$$\sum_{v \in N} I_v = 0, \quad \sum_{v \in N} \dot{I}_v = 0, \quad N = \{a, b, \dots, p\} \tag{2.105}$$

Die Gleichungen (2.104), (2.105) müssen gemeinsam nach dem Referenzpotential U_{REF} und dem Vektor der Ableitungen $\dot{\mathbf{I}}$ aufgelöst werden. Aus diesen Gleichungen wird ein Sparse-Tableau formuliert. Dabei entstehen aus den elementaren Matrizen und Vektoren die größeren Matrizen $\hat{\mathbf{R}}^*$, $\hat{\mathbf{T}}$ und Vektoren $\hat{\mathbf{U}}^*$, $\hat{\mathbf{X}}$, $\hat{\mathbf{U}}$ (Bild 2.38).

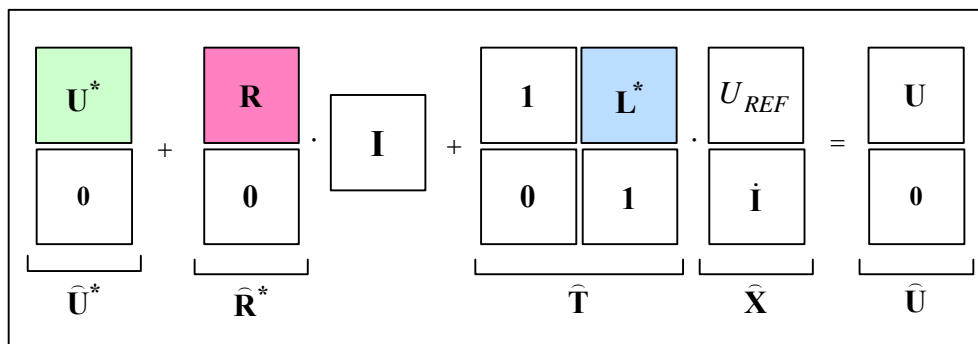


Bild 2.38: Sparse-Tableau auf der obersten Ebene

Die Tableau-Matrizen Gleichung in Bild 2.38 kann man umformen als

$$\hat{\mathbf{U}}^* + \hat{\mathbf{R}}^* \cdot \mathbf{I} + \hat{\mathbf{T}} \cdot \hat{\mathbf{X}} = \hat{\mathbf{U}}. \tag{2.106}$$

Dabei ist $\hat{\mathbf{X}}$ der Vektor der unbekanntenen Ableitungen der Ströme und des Referenzpotentials. Der Vektor $\hat{\mathbf{X}}$ hat $p+1$ Elemente. Der Vektor der Eingangsströme \mathbf{I} besteht aus den

Zustandsgrößen und ist deshalb zu jeder Zeit bekannt. Der Vektor der Spannungen an den Pins $\hat{\mathbf{U}}$ ist bekannt. Der Vektor $\hat{\mathbf{U}}$ hat $p+1$ Elemente, und der Vektor \mathbf{I} hat p Elemente.

Die Matrix $\hat{\mathbf{R}}^*$ hat die Dimension $p \times (p+1)$. Die Matrix $\hat{\mathbf{T}}$ ist eine $(p+1) \times (p+1)$ Quadratmatrix.

Die Gleichung (2.75) wird erweitert:

$$\begin{pmatrix} \hat{U}_a^* \\ \hat{U}_b^* \\ \vdots \\ \hat{U}_p^* \\ 0 \end{pmatrix} + \begin{pmatrix} \hat{R}_{aa}^* & \hat{R}_{ab}^* & \cdots & \hat{R}_{ap}^* \\ \hat{R}_{ba}^* & \hat{R}_{bb}^* & \cdots & \hat{R}_{bp}^* \\ \vdots & \vdots & \ddots & \vdots \\ \hat{R}_{pa}^* & \hat{R}_{pb}^* & \cdots & \hat{R}_{pp}^* \\ 0 & 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} I_a \\ I_b \\ \vdots \\ I_p \end{pmatrix} + \begin{pmatrix} 1 & \hat{L}_{aa}^* & \hat{L}_{ab}^* & \cdots & \hat{L}_{ap}^* \\ 1 & \hat{L}_{ba}^* & \hat{L}_{bb}^* & \cdots & \hat{L}_{bp}^* \\ 1 & \vdots & \vdots & \ddots & \vdots \\ 1 & \hat{L}_{pa}^* & \hat{L}_{pb}^* & \cdots & \hat{L}_{pp}^* \\ 0 & 1 & 1 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} U_{REF} \\ \dot{I}_a \\ \dot{I}_b \\ \vdots \\ \dot{I}_p \end{pmatrix} = \begin{pmatrix} U_a \\ U_b \\ \vdots \\ U_p \\ 0 \end{pmatrix}. \quad (2.107)$$

Die Berechnung des Stromes am Pin mit der offenen Induktivität erfolgt durch die Integration des Stromes. Für den Strom am Pin ohne die offene Induktivität werden nur die algebraischen Gleichungen verwendet.

Die Gleichung (2.107) entspricht der Voraussetzung, dass alle Pins der Komponente in Bild 2.37 offene Induktivitäten haben. Nimmt man an, dass der Pin b keine offene Induktivität enthält, dann sind die Parameter $\hat{L}_{ba}^*, \dots, \hat{L}_{bp}^*$ gleich Null. Für die Spannung U_b wird eine algebraische Gleichung statt einer Differenzialgleichung formuliert. Der Strom I_b wird keine Zustandsgröße und muss aufgelöst werden. Dafür muss er in den Vektor $\hat{\mathbf{X}}$ eingesetzt werden. Die Ableitung des Stromes \dot{I}_b wird aus dem Vektor $\hat{\mathbf{X}}$ eliminiert. Man muss die entsprechenden Manipulationen auch mit den Matrizen $\hat{\mathbf{R}}^*$, $\hat{\mathbf{T}}$ durchführen. Die Änderungen in den Matrizen und Vektoren werden nach dem folgenden Algorithmus *VarCalc* ermittelt:

Schritt 1

Die Pins mit der offenen Induktivität werden bestimmt. Dafür werden die Spalten der Matrix $\tilde{\mathbf{L}}^*$ gesucht, die nur Nullen enthalten (Bild 2.39). Das ist die Spalte b .

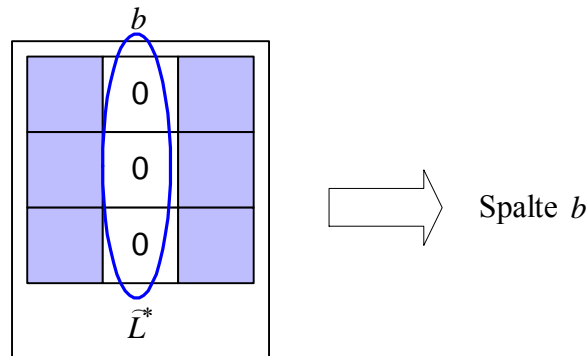


Bild 2.39: Definition der Spalte mit den Nullen (Schritt 1)

Schritt 2

Nach der Elimination der Ableitungen \dot{I}_b wird die Gleichung $\sum_{v \in N} \dot{I}_v = 0$ für die Ableitungen

durch die Stromgleichung $\sum_{v \in N} I_v = 0$ ersetzt. Dafür werden die Elemente mit den Nullen in

der letzten Zeile der Matrix $\hat{\mathbf{R}}^*$ nach den Einsen geändert. Die Elemente mit den Einsen in der letzten Zeile der Matrix $\hat{\mathbf{T}}$ werden nach den Nullen geändert (Bild 2.40).

Schritt 3

Die Zeile b des Vektors \mathbf{I} wird herausgenommen und an der Stelle $(b+1)$ des Vektors $\hat{\mathbf{X}}$ eingesetzt. Die Spalte $(b+1)$ der Matrix $\hat{\mathbf{R}}^*$ wird herausgenommen und in die Spalte $(b+1)$ der Matrix $\hat{\mathbf{T}}$ eingesetzt (Bild 2.41).

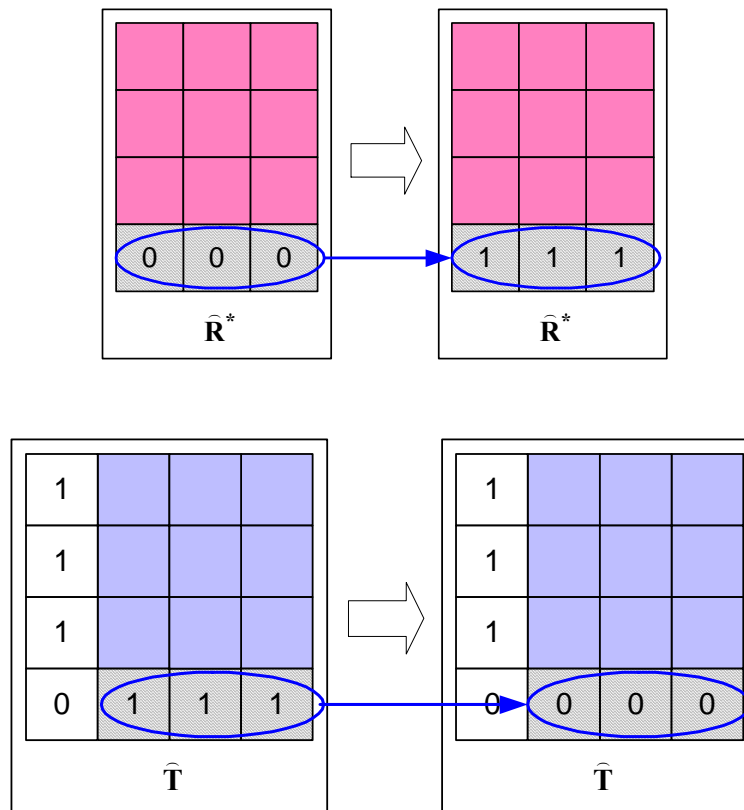


Bild 2.40: Änderung der Matrizen $\hat{\mathbf{R}}^*$ und $\hat{\mathbf{T}}$ (Schritt 2)

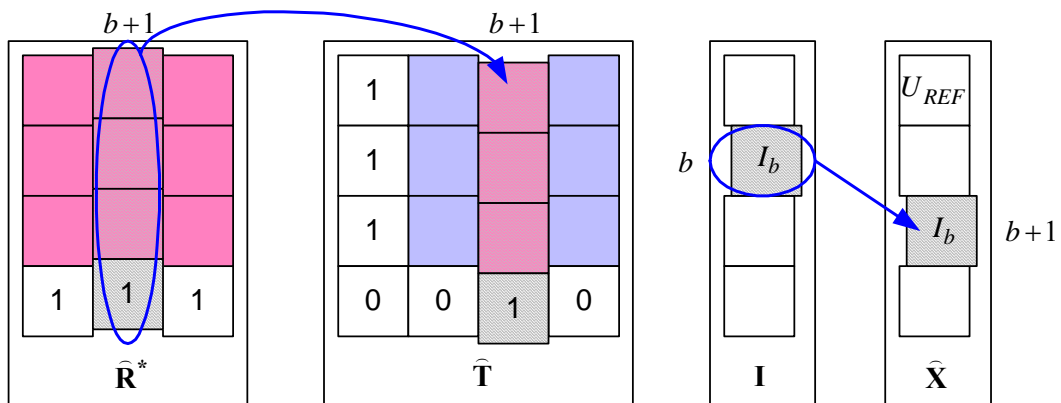


Bild 2.41: Übertragung der I_b (Schritt 3)

Nach der Anwendung ergibt sich der Änderungsalgorithmus für die Matrixgleichung (2.107):

$$\begin{pmatrix} \widehat{U}_a^* \\ \widehat{U}_b^* \\ \vdots \\ \widehat{U}_p^* \\ 0 \end{pmatrix} + \begin{pmatrix} \widehat{R}_{aa}^* & \widehat{R}_{ac}^* & \cdots & \widehat{R}_{ap}^* \\ \widehat{R}_{ba}^* & \widehat{R}_{bc}^* & \cdots & \widehat{R}_{bp}^* \\ \vdots & \vdots & \ddots & \vdots \\ \widehat{R}_{pa}^* & \widehat{R}_{pc}^* & \cdots & \widehat{R}_{pp}^* \\ 1 & 1 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} I_a \\ I_c \\ \vdots \\ I_p \end{pmatrix} + \begin{pmatrix} 1 & \widehat{L}_{aa}^* & \widehat{R}_{ab}^* & \cdots & \widehat{L}_{ap}^* \\ 1 & \widehat{L}_{ba}^* & \widehat{R}_{bb}^* & \cdots & \widehat{L}_{bp}^* \\ 1 & \vdots & \vdots & \ddots & \vdots \\ 1 & \widehat{L}_{pa}^* & \widehat{R}_{pb}^* & \cdots & \widehat{L}_{pp}^* \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} U_{REF} \\ \dot{I}_a \\ I_b \\ \vdots \\ \dot{I}_p \end{pmatrix} = \begin{pmatrix} U_a \\ U_b \\ \vdots \\ U_p \\ 0 \end{pmatrix} \quad (2.108)$$

oder in der Matrixform:

$$\widehat{\mathbf{U}}^* + \widetilde{\mathbf{R}}^* \cdot \widetilde{\mathbf{I}} + \widetilde{\mathbf{T}} \cdot \widetilde{\mathbf{X}} = \widehat{\mathbf{U}}. \quad (2.109)$$

Dabei ist $\widetilde{\mathbf{R}}^*$ die geänderte Matrix $\widehat{\mathbf{R}}^*$, $\widetilde{\mathbf{I}}$ ist der geänderte Vektor \mathbf{I} , $\widetilde{\mathbf{T}}$ ist die geänderte Matrix $\widehat{\mathbf{T}}$ und $\widetilde{\mathbf{X}}$ ist der geänderte Vektor $\widehat{\mathbf{X}}$.

Die Matrixgleichung (2.109) muss nach $\widetilde{\mathbf{X}}$ auflöst werden. Umgeformt ergibt sich:

$$\widetilde{\mathbf{T}} \cdot \widetilde{\mathbf{X}} = \widehat{\mathbf{U}} - \widehat{\mathbf{U}}^* + \widetilde{\mathbf{R}}^* \cdot \widetilde{\mathbf{I}}. \quad (2.110)$$

Nach der Anwendung der Gauß-Elimination auf die Gleichung (2.108) erhält man

$$\begin{aligned} \widetilde{\mathbf{X}} &= \widetilde{\mathbf{B}}, \\ \widetilde{\mathbf{B}} &= f(\widehat{\mathbf{U}}, \widetilde{\mathbf{I}}). \end{aligned} \quad (2.111)$$

Dabei ist $\widetilde{\mathbf{B}}$ der Vektor der Lösungsausdrücke.

Ein entsprechendes Maple - Programm befindet sich im Anhang B.

Beispiel 2.4

Das Netzwerk aus dem Beispiel 2.3 ist an die Spannungsquelle VS und Masse angeschlossen (Bild 2.42). Dabei gilt $U_{Pa} = U_{VS}$, $U_{Pb} = 0$. Die Komponente hat $p = 2$ Pins.

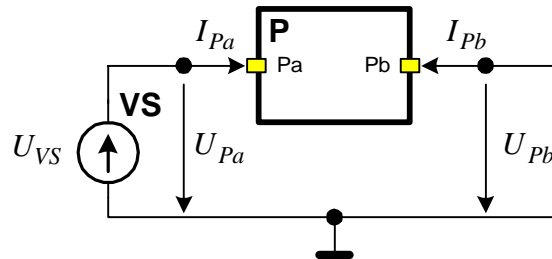


Bild 2.42: Definition der Eingangsspannungen

Aus der Tabelle 2.12 sind die Schnittstellenparameter des Blocks P zu entnehmen:

$$\mathbf{U}^* = \begin{pmatrix} 0 \\ \frac{L_2 L_3}{L_2 + L_3} (I_{L_2}(0) + I_{L_3}(0)) \end{pmatrix}, \quad \mathbf{R}^* = \begin{pmatrix} R_1 & 0 \\ 0 & R_4 \end{pmatrix}, \quad \mathbf{L}^* = \begin{pmatrix} 0 & 0 \\ 0 & \frac{L_2 L_3}{L_2 + L_3} \end{pmatrix}. \quad (2.112)$$

Die Spannungs- und Stromgleichungen (2.104) und (2.105) für die Komponente P mit zwei Pins ergeben sich als:

$$\begin{pmatrix} 0 \\ \frac{L_2 L_3}{L_2 + L_3} (I_{L_2}(0) + I_{L_3}(0)) \end{pmatrix} + \begin{pmatrix} R_1 & 0 \\ 0 & R_4 \end{pmatrix} \cdot \begin{pmatrix} I_{Pa} \\ I_{Pb} \end{pmatrix} + \begin{pmatrix} U_{REF} \\ U_{REF} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & \frac{L_2 L_3}{L_2 + L_3} \end{pmatrix} \cdot \begin{pmatrix} \dot{I}_{Pa} \\ \dot{I}_{Pb} \end{pmatrix} = \begin{pmatrix} U_{VS} \\ 0 \end{pmatrix}. \quad (2.113)$$

$$I_{Pa} + I_{Pb} = 0, \quad \dot{I}_{Pa} + \dot{I}_{Pb} = 0.$$

Die Sparse-Tableau Formulierung aus (2.113) ist:

$$\begin{aligned}
& \begin{pmatrix} 0 \\ \frac{L_2 L_3}{L_2 + L_3} (I_{L_2}(0) + I_{L_3}(0)) \\ 0 \end{pmatrix} + \begin{pmatrix} R_1 & 0 \\ 0 & R_4 \\ 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} I_{Pa} \\ I_{Pb} \end{pmatrix} + \\
& + \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & \frac{L_2 L_3}{L_2 + L_3} \\ 0 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} U_{REF} \\ \dot{I}_{Pa} \\ \dot{I}_{Pb} \end{pmatrix} = \begin{pmatrix} U_{VS} \\ 0 \\ 0 \end{pmatrix}.
\end{aligned} \tag{2.114}$$

Die Spalte 1 in der Matrix \mathbf{L}^* (die Gleichung (2.112)) enthält nur Nullen. Der Pin Pa ist ein Pin ohne offene Induktivität. Für den Pin Pa wird nur der Strom I_{Pa} definiert und berechnet.

Die Ableitung \dot{I}_{Pa} muss aus der Matrixgleichung (2.114) eliminiert werden.

Die Änderung der Matrizen nach dem vorgeschlagenen Algorithmus ergibt für die Gleichung (2.109):

$$\begin{aligned}
& \begin{pmatrix} 0 \\ \frac{L_2 L_3}{L_2 + L_3} (I_{L_2}(0) + I_{L_3}(0)) \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ R_4 \\ 1 \end{pmatrix} \cdot (I_{Pb}) + \\
& + \begin{pmatrix} 1 & R_1 & 0 \\ 1 & 0 & \frac{L_2 L_3}{L_2 + L_3} \\ 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} U_{REF} \\ I_{Pa} \\ \dot{I}_{Pb} \end{pmatrix} = \begin{pmatrix} U_{VS} \\ 0 \\ 0 \end{pmatrix}.
\end{aligned} \tag{2.115}$$

Auf der obersten Ebene wird die Ableitung des Stroms \dot{I}_{Pb} berechnet, der Strom I_{Pb} wird nicht integriert, deshalb braucht man den Anfangswert des Stroms $I_{Pb}(0)$ nicht.

Der Anfangswert wird auf Null gesetzt: $(I_{L_2}(0) + I_{L_3}(0)) = 0$.

Die Auflösung (2.115) mit der Gauß-Elimination ergibt:

$$\begin{aligned}
 U_{REF} &= U_{VS} - R_1 I_{Pa}, \\
 I_{Pa} &= \left[R_4 I_{Pb} + U_{VS} + \frac{L_1 L_2}{L_1 + L_2} \dot{I}_{Pb} \right] \cdot \frac{1}{R_1}, \\
 \dot{I}_{Pb} &= \left[-U_{VS} - (R_4 + R_1) I_{Pb} \right] \cdot \frac{L_1 + L_2}{L_1 L_2}.
 \end{aligned} \tag{2.116}$$

Die Größen werden nacheinander berechnet. Zuerst wird der Wert von \dot{I}_{Pb} ermittelt, der dann in der Gleichung für I_{Pa} eingesetzt wird. Der berechnete Strom I_{Pa} wird in die Gleichung für U_{REF} eingesetzt.

2.5.3 Berechnung der Unbekannten in der Komponente

Man betrachtet eine Komponente mit p Pins (Bild 2.44). Man nimmt an, dass h Pins offene Induktivitäten haben (sie sind für eine bessere Anschaulichkeit in Bild 2.43 links dargestellt). Die übrigen $p-h$ Pins enthalten keine offenen Induktivitäten (sie sind in Bild 2.43 rechts dargestellt).

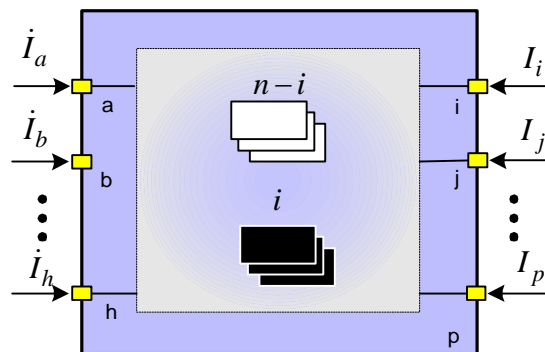


Bild 2.43: Komponente mit i inneren Induktivitäten und p Pins

Durch die Pins fließen h Ströme $I_a \dots I_h$ mit ihren Ableitungen $\dot{I}_a \dots \dot{I}_h$ und $p-h$ Ströme ohne Ableitungen $I_i \dots I_p$, insgesamt ergeben sich p Schnittstellenströme. Die Komponente erhält von ihrem Parentblock die Werte der h berechneten Ableitungen $\dot{I}_a \dots \dot{I}_h$, die Werte der $p-h$ Ströme $I_i \dots I_p$ und das Referenzpotential U_{REF} . p Schnittstellenspannungen

$U_a \dots U_p$ stehen ebenfalls zur Verfügung. Diese wurden vorher bereits im Parentblock berechnet. Auf der obersten Ebene erhält der Parentblock die vordefinierten Schnittstellenspannungen. Die bekannten und unbekanntenen Schnittstellenströme und ihre Ableitungen sind in Tabelle 2.13 zusammengefasst.

Tabelle 2.13: Schnittstellenströme des Blocks

Variablen (Anzahl)	erhalten vom Parentblock (Anzahl)	zu berechnen (Anzahl)
Schnittstellenströme (p)	$I_i \dots I_p (p-h)$	$I_a \dots I_h (h)$
Ableitungen der Schnittstellenströme (h)	$\dot{I}_a \dots \dot{I}_h (h)$	–

Man nimmt an, dass die Komponente Subblöcke mit insgesamt n Zweigen und m Knoten enthält. Unter diesen sind i Zweigströme der Subblocks $I_{SUB.a} \dots I_{SUB.i}$ bekannt als Zustandsgröße der inneren Induktivitäten. Die $n-i$ Zweigströme $I_{SUB.j} \dots I_{SUB.n}$ sind unbekannt und müssen berechnet werden. Auch müssen die Werte der i Ableitungen der Zweige mit den Induktivitäten $\dot{I}_{SUB.a} \dots \dot{I}_{SUB.i}$ berechnet werden. Sie werden zu ihren Subblocks für die Integration der neuen Werte der Ströme geschickt. Die bekannten und unbekanntenen Zweigströme und ihre Ableitungen sind in Tabelle 2.14 zusammengefasst.

Tabelle 2.14: Zweigströme des Blocks

Variablen (Anzahl)	erhalten von den Subblocks (Anzahl)	zu berechnen (Anzahl)
Zweigströme (n)	$I_{SUB.a} \dots I_{SUB.i} (i)$	$I_{SUB.j} \dots I_{SUB.n} (n-i)$
Ableitungen der Zweigströme (i)	–	$\dot{I}_{SUB.a} \dots \dot{I}_{SUB.i} (i)$

Die Matrixgleichung im Bildbereich (2.75) im Abschnitt 2.4.4 wird für die Berechnung der Schnittstellenparameter verwendet. Die Matrixgleichung ist so formuliert, dass der Vektor der Unbekannten \mathbf{X} nur die Zweigströme als Unbekannte enthält:

$$\mathbf{A}_{PE} \cdot \mathbf{I}_P + \mathbf{U}_{REF} + \mathbf{T} \cdot \mathbf{X} = \mathbf{U}_{OE},$$

$$\mathbf{X} = (\mathbf{I}_Z \quad \mathbf{P}_Z)^T, \quad (2.117)$$

$$\mathbf{X} = (I_{SUB.a} \quad \cdots \quad I_{SUB.p} \quad U_{N1} \quad \cdots \quad U_{Nm})^T.$$

Dabei ist $I_{SUB.v}$ der Strom des Zweiges v der betrachteten Komponente, U_{Nv} ist das Potential des Knotens v der Komponente.

Die Ausdrücke für die Ableitungen der Ströme werden nach der symbolischen Auflösung der Ströme durch die Multiplikationen mit s erhalten. Es wird angenommen, dass alle Schnittstellenströme bekannt sind. Die Realisierung der Reihenfolge der Berechnung, wie im Abschnitt 2.5.1 dargestellt wurde, muss durch die Änderung der Matrizen erfolgen. Die Variablen (die Ströme und die Spannungen oder die Potentiale), die unbekannt sind und berechnet werden sollen, werden in Vektor \mathbf{X} eingesetzt. Die Variablen, die man vom Parentblock oder von den Subblöcken erhält, werden in Vektor \mathbf{I}_P eingesetzt. Die Änderungen werden nach dem folgenden Algorithmus *TopMod* vorgenommen:

Schritt 1

Der Algorithmus wird am Beispiel nur einer Zustandgröße $I_{SUB.v}$ (Induktivitätsstrom) betrachtet. Man erweitert den Algorithmus aber auf alle Zustandsgrößen. Im Vektor \mathbf{X} wird die Zustandgröße bestimmt. Dafür analysiert man den Parameter $L_{SUB.v}^*$ des Subblocks, der den hier fließenden Strom definiert. Falls $L_{SUB.v}^* \neq 0$ gilt, ist der Strom eine Zustandgröße. Nach der Analyse erhält man den Index des Elements des Vektors \mathbf{X} (Bild 2.44).

Schritt 2

Aus der Matrixgleichung (2.75) wird die Gleichung eliminiert, die die Zustandgröße bestimmt. Die Zustandgröße wird in den Vektor der Bekannten \mathbf{I}_P eingesetzt.

Die Zeile v des Vektors \mathbf{X} wird in den Vektor \mathbf{I}_P an der Stelle $p+1$ hinzugefügt, die Spalte v der Matrix \mathbf{T} wird in der Matrix \mathbf{A}_{PE} an der Stelle $p+1$ hinzugefügt (Bild 2.45). Aus den Matrizen \mathbf{A}_{PE} , \mathbf{T} und Vektoren \mathbf{U}_{OE} , \mathbf{U}_{REF} wird die Zeile v eliminiert.

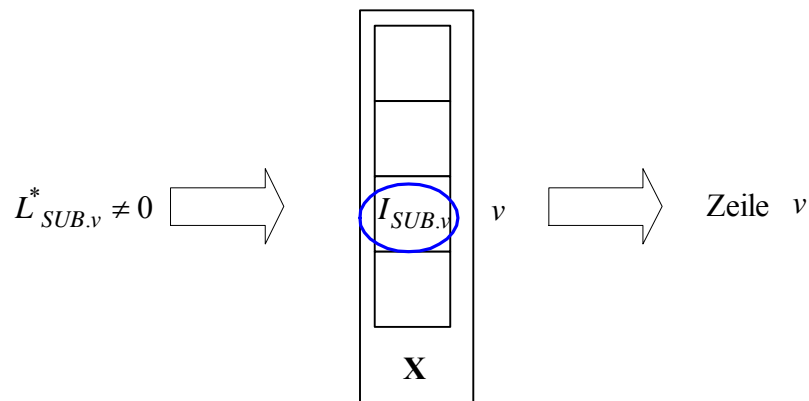


Bild 2.44: Definition der Zustandsgrößen (Schritt 1)

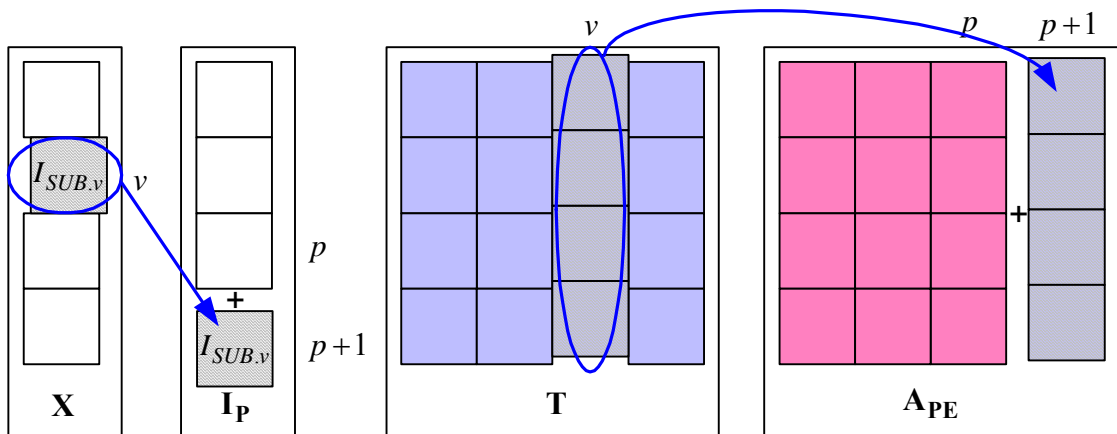


Bild 2.45: Einsetzen der Zustandsgrößen (Schritt 2)

Schritt 3

Man betrachtet die Schnittstellenströme der Komponente. Wenn der Strom I_v zum Pin mit der offenen Induktivität ($L_v^* \neq 0$) gehört, wird der Schnittstellenstrom I_v im Vektor \mathbf{I}_P mit der Schnittstellenspannung U_v im Vektor \mathbf{X} umgetauscht. Die Spalte v der Matrix \mathbf{A}_{PE} wird mit der Spalte g in der Matrix \mathbf{T} umgetauscht (Bild 2.46).

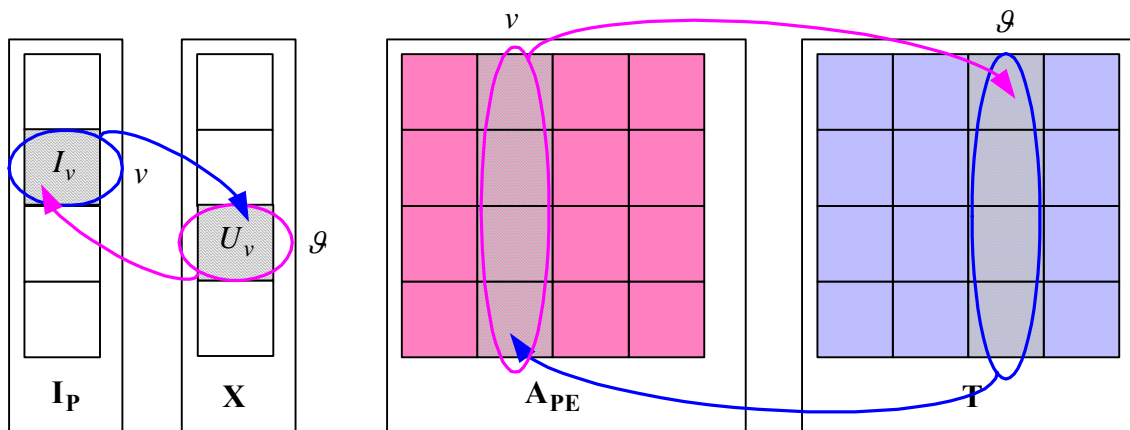


Bild 2.46: Umtauschen der Schnittstellengrößen (Schritt 3)

Die nach dem Algorithmus geänderte Matrixgleichung (2.75) ergibt sich wie folgt:

$$\underline{\underline{A}}_{PE} \cdot \underline{\underline{I}}_P + \underline{\underline{U}}_{REF} + \underline{\underline{T}} \cdot \underline{\underline{X}} = \underline{\underline{U}}_{OE} \quad (2.118)$$

oder umgeformt:

$$\underline{\underline{T}} \cdot \underline{\underline{X}} = \underline{\underline{U}}_{OE} - \underline{\underline{A}}_{PE} \cdot \underline{\underline{I}}_P - \underline{\underline{U}}_{REF}. \quad (2.119)$$

Die Auflösung nach der Gauß-Elimination nach $\underline{\underline{X}}$ ergibt:

$$\underline{\underline{X}} = \underline{\underline{B}}. \quad (2.120)$$

Ein Beispiel für die Anwendung des Algorithmus wird in Kapitel 3 betrachtet.

3 Spezielle Punkte der Modellierung und Ablauf der Modellierung

3.1 Behandlung der Schaltvorgänge

Ein wichtiger Teil der Modellierung eines mechatronischen Systems ist die Behandlung der Schaltvorgänge. Ein mechatronisches System enthält meist Schaltungselemente wie Schalter oder Transistoren im Schaltbetrieb. Man betrachtet den Schalter als einen von außen gesteuerten Widerstand mit zwei Zuständen (Abschnitt 2.3.3). Im geschlossenen Zustand ist der Widerstand des Schalters sehr klein ($0,01 \dots 1 \Omega$), im geöffneten Zustand ist der Widerstand dagegen sehr groß ($1 \dots 100 \text{ M}\Omega$). Falls eine modellierte Komponente innere Induktivitäten hat, kann die Komponente mit dem geöffneten Schalter eine sehr kleine Zeitkonstante haben [JUS94].

Der Ausgleichsprozess der RL -Reihenschaltung aus Bild 3.1 wird zum Beispiel mit einer Differentialgleichung erster Ordnung beschrieben [EDM90]:

$$L \cdot \dot{I} + R \cdot I = U. \quad (3.1)$$

Dabei ist R der Widerstand des Schalters.

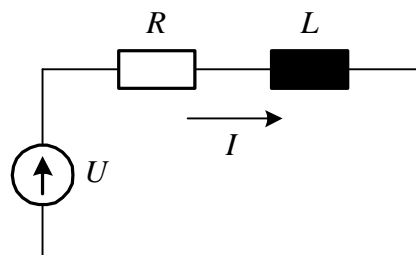


Bild 3.1: RL -Reihenschaltung

Die Zeitkonstante τ ist umgekehrt proportional zu R :

$$\tau = L/R \quad (3.2)$$

Für die typischen Werte $R = 1 \text{ M}\Omega$, $L = 1 \text{ mH}$ ist $\tau = 1 \text{ ns}$.

Für die korrekte Integration der Ströme müsste der Integrationsschritt 10...100 mal kleiner sein. Die Simulationszeit bei der Integration mit dem kleinen Integrationsschritt steigt deutlich. Bild 3.2 zeigt die Steigerung der Simulationszeit in der Abhängigkeit vom Integrationsschritt bei der Simulation des elektrischen Kreises aus Beispiel 2.3.

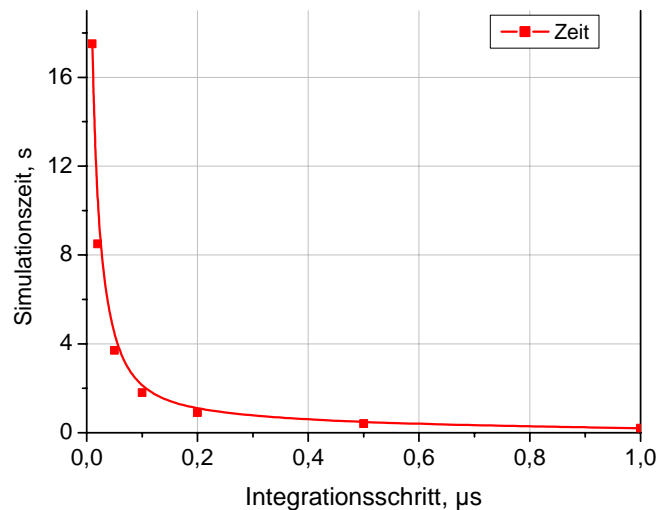


Bild 3.2: Abhängigkeit der Simulationszeit vom Integrationsschritt

Die Simulation wurde an einem Rechner mit dem Prozessor Athlon64 – 3200 durchgeführt. Die Vergrößerung des Integrationsschritts führt schnell zu einem großen Fehler und zur Nichtkonvergenz bei der Simulation eines Systems höherer Ordnung [KIE94]. Eine mögliche Lösung ist die Anwendung adaptiver Integrationsschritte. Der Zeitschritt wird dynamisch abhängig von der Netzaktivität, der Änderung der Spannung oder des Stroms gesteuert. SPICE verwendet diesen Algorithmus [KIE94]. Bei einer Steigerung der Ableitung oder der Netzaktivität wird der Zeitschritt kleiner eingestellt (Bild 3.3). In der Periode niedriger Netzaktivität oder kleiner Änderung der Spannung oder des Stroms vergrößert der Algorithmus der Zeitschrittsteuerung den Zeitschritt.

Ein anderer Weg ist die separate Behandlung der Zustände, wobei sehr kleine Zeitkonstanten von vorne herein vermieden werden. Dabei werden die Sonderfälle der Differentialgleichung berücksichtigt, die man flexibel behandeln kann. Wenn der Schalter geschlossen ist, dann ist die Zeitkonstante groß und es gibt keine Schwierigkeiten bei der Integration der Differentialgleichung (3.1).

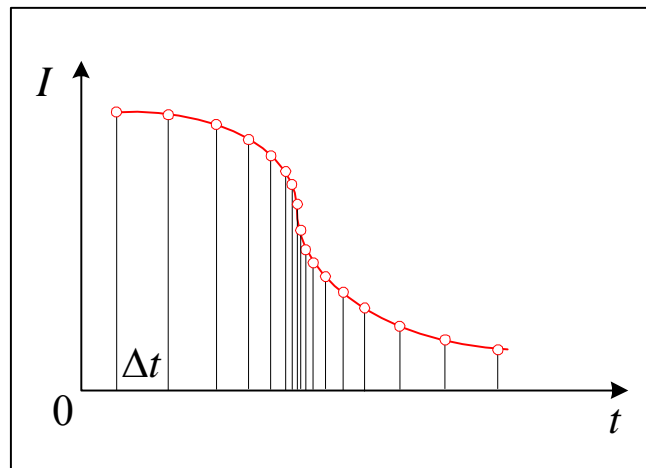


Bild 3.3: Dynamische Änderung des Zeitschrittes bei der Simulation

Bei geöffnetem Schalter wird die Zeitkonstante sehr klein und die berechnete Ableitung der Zustandsgröße wird sehr groß. Der Ausgleichsprozess bei der schnellen Änderung der Zustandsgröße kann vernachlässigt werden. Der Term, der die Ableitung der Zustandsgröße enthält und die Dynamik der Zustandsgröße bestimmt, wird aus der Gleichung eliminiert:

$$L \cdot \dot{I} + R \cdot I = U \Rightarrow R \cdot I = U, \dot{I} = 0. \quad (3.3)$$

Statt einer Differenzialgleichung wird eine algebraische Gleichung verwendet und so kann man auf den Integrator verzichten.

Der Wert der Zustandsgröße muss nach der Vernachlässigung des Ausgleichsprozesses richtig berechnet werden. Ein idealer Schalter schaltet sofort ohne Verzögerung zum Zeitpunkt t_S um (Bild 3.4). Der Widerstand des Zweiges springt sofort zu einem sehr großen Wert. Wenn der Zweig eine Induktivität enthält, dann wird die Zeitkonstante des Ausgleichsprozesses sehr klein, und der Strom verringert sich stark für die Zeit T im Bereich $I(t_S)$ bis $I(t_S + T)$ (Bild 3.4a). Nach der Eliminierung des schnellen Ausgleichsprozesses wird die stationäre Lösung ab dem Zeitpunkt $t_S + T$ berechnet. Der Ausgleichsprozess des Stromes wird als „Sprung“ von $I(t_S)$ bis $I(t_S + T)$ auf die Größe $I_O(t_S)$ modelliert (Bild 3.4b).

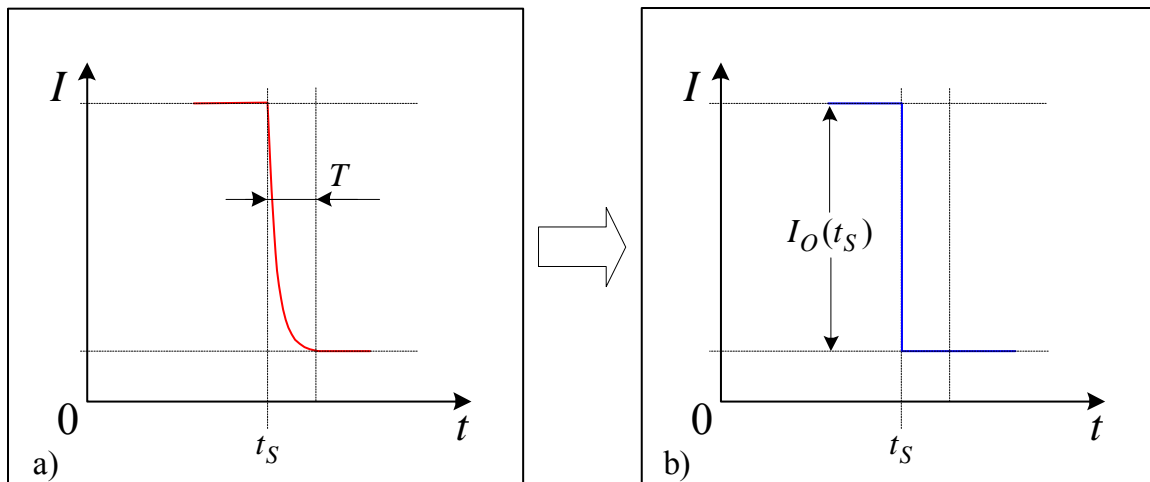


Bild 3.4: Verringerung des Ausgleichsprozesses während der Umschaltung

Die stationäre Lösung nach der Umschaltung aus (3.3) ergibt sich zu

$$I(t_S) = \frac{U}{R}. \quad (3.4)$$

Aus physikalischem Grund wird die in der Spule gespeicherte Energie im Schalter abgebaut:

$$R = R_{OFF}, R_{OFF} \rightarrow \infty, I(t_S) \rightarrow 0. \quad (3.5)$$

Es gibt aber auch eine zweite Variante, dabei kann die Energie der Spule in ein anderes reaktives Element (z.B. Kondensator) übertragen werden. Dies ist möglich, falls die Umschaltung durch die Diode (oder ein anderes von innen gesteuertes Element) verwendet wird. Die gesperrte Diode muss zum Abbauen des Induktivitätsstroms geöffnet werden. In diesem Fall wird, für die korrekte Behandlung des Ausgleichsprozesses, der Wert der Zustandsgröße analysiert. Wenn der Wert der Zustandsgröße (des Stromes in diesem Beispiel) genügend klein wird, dann korrespondiert die stationäre Lösung (3.4) mit der Bedingung (3.5). Wenn der Wert aber groß ist, dann muss die Diode geöffnet werden, um die gespeicherte Energie der Spule abzubauen:

$$I(t_S) = \frac{U}{R_{ON}}, (R = R_{OFF} \rightarrow R_{ON}). \quad (3.6)$$

3.2 Änderung der Reihenfolge der Berechnung der Unbekannten

Die Matrixgleichung (2.75) für die Berechnung der Unbekannten ergibt sich wie folgt:

$$\mathbf{A}_{PE} \cdot \mathbf{I}_P + \mathbf{U}_{REF} + \mathbf{T} \cdot \mathbf{X} = \mathbf{U}_{OE}. \quad (3.7)$$

Der Vektor \mathbf{X} definiert die Reihenfolge der Berechnung der Unbekannten in der Strukturkomponente. Für die optimale und korrekte Behandlung der Schaltvorgänge nach Abschnitt 3.1 müssen die Induktivitätsströme wie möglichst unten im Vektor \mathbf{X} stehen. Auch die selten abgefragten Knotenspannungen müssen möglichst oben stehen.

Die Reihenfolge der Unbekannten im Vektor \mathbf{X} kann durch die Multiplikation mit der Permutationsmatrix \mathbf{P}_i geändert werden:

$$\mathbf{A}_{PE} \cdot \mathbf{I}_P + \mathbf{U}_{REF} + \mathbf{T} \cdot \mathbf{P}_i \cdot \mathbf{P}_i^T \cdot \mathbf{X} = \mathbf{U}_{OE}. \quad (3.8)$$

Dabei gilt $\dim(\mathbf{P}_i) = \dim(\mathbf{T})$. Jede Zeile der Matrix \mathbf{P}_i besteht aus Nullen und einer Eins. Der Spaltenindex der Eins definiert die Stelle des Elements des Vektors \mathbf{X} vor der Permutation, der Zeilenindex der Eins definiert die Stelle im Vektor \mathbf{X} nach der Permutation.

Der Vektor \mathbf{X} wird so permutiert, dass die Zustandsströme an das Ende des Vektors \mathbf{X} umgestellt werden.

Die Anwendung der Funktion *GaussCalc* auf die Matrixgleichung (3.8) ergibt den Vektor der Lösungsausdrücke \mathbf{B} für den Vektor \mathbf{X} :

$$\begin{aligned} \mathbf{P}_i^T \cdot \mathbf{X} &= \mathbf{U}_{OE} - (\mathbf{A}_{PE} \cdot \mathbf{I}_P + \mathbf{U}_{REF} + \mathbf{T} \cdot \mathbf{P}_i), \\ \mathbf{B} &= \mathbf{GaussCalc} \left[\mathbf{U}_{OE} - (\mathbf{A}_{PE} \cdot \mathbf{I}_P + \mathbf{U}_{REF} + \mathbf{T} \cdot \mathbf{P}_i) \right], \\ \mathbf{P}_i^T \cdot \mathbf{X} &= \mathbf{B}. \end{aligned} \quad (3.9)$$

3.3 Ablauf der Modellierung

Im Ablauf der Modellierung werden vier verschiedene funktionale Formen der Operationen berücksichtigt (Bild 3.5).

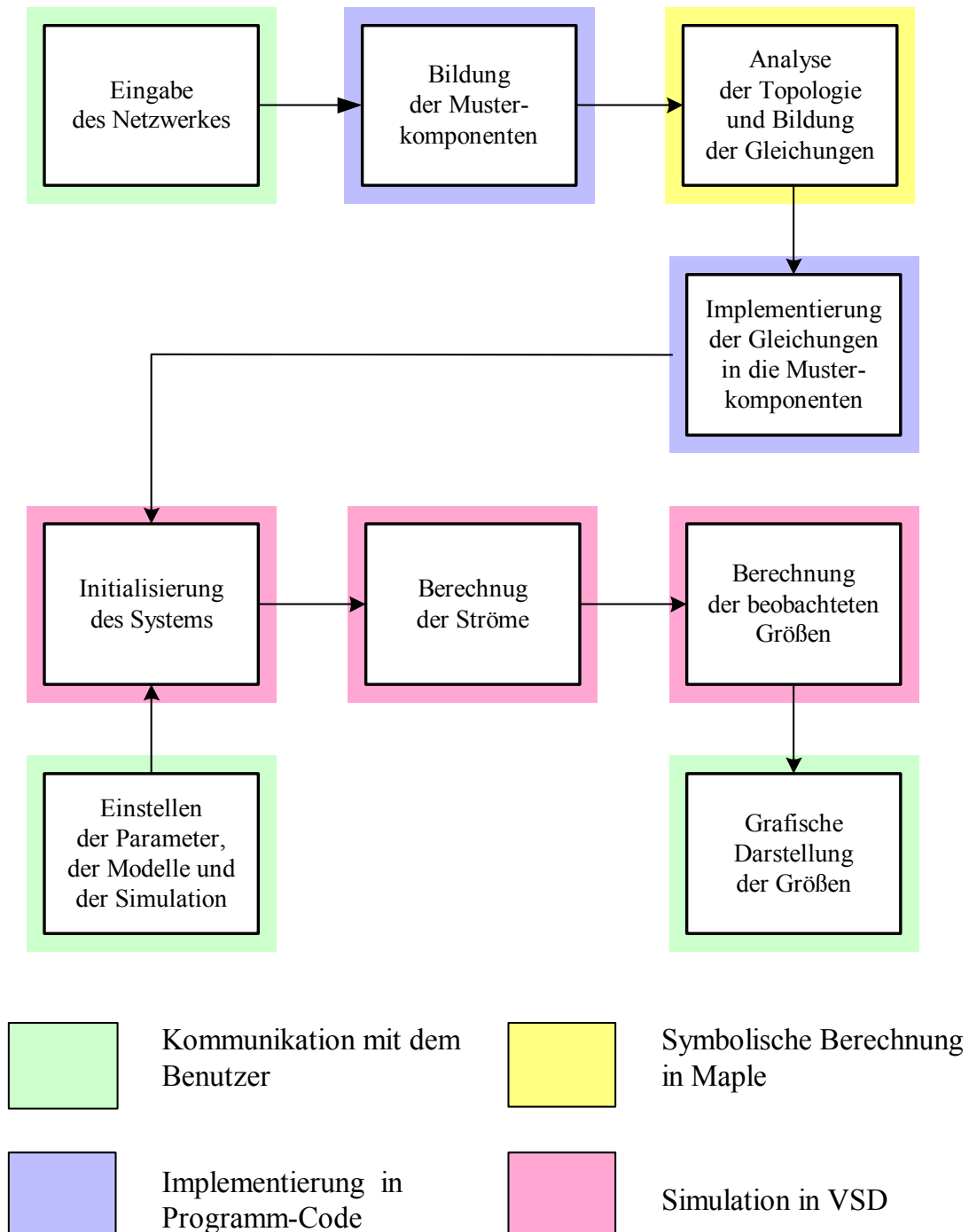


Bild 3.5: Ablauf der Modellierung

Diese Formen sind:

- Die Operationen der Kommunikation mit dem Benutzer
- Die Implementierung der bestimmten Elemente des Modells in den Programm-Code der Modell-Komponente
- Symbolische Berechnung in Maple
- Simulation des komponentenorientierten Modells in VSD

Eingabe des Netzwerkes

Die Eingabe des elektrischen Netzwerkes wird in Kapitel 4 ausführlich betrachtet. Es gibt verschiedene Formen der Eingabe des Netzwerkes. Die Struktur des Netzwerkes und die Parameter der Elemente können aus der Netzliste im SPICE - Format oder aus dem XML - basierten eigenen Format importiert werden. Die andere Möglichkeit ist die visuelle Drag & Drop Technologie in Visio für die Einbindung der Basiskomponente in die hierarchischen Strukturblöcke. Die Basisblöcke werden aus den Bibliotheken der Standardkomponenten genommen.

Bildung der Musterkomponenten

Aus der Analyse der Topologie in der Netzliste oder verbundener Basisobjekte in Visio werden die Musterkomponenten des Modells des modellierten Netzwerkes gebildet. Die Grundbeziehungen zwischen den Komponenten und die Schnittstellen werden implementiert.

Bildung der Gleichungen

Aus der syntaktischen Analyse des Netzwerkes und der symbolischen Berechnungen in Maple ergeben sich die Gleichungen, die Parameter und die Zustände, die das Modell des Netzwerkes bestimmen.

Implementierung der Gleichungen in die Musterkomponenten

Die Musterkomponenten des Modells des Netzwerkes werden mit den symbolisch berechneten Gleichungen, Parametern und Zuständen erzeugt. Die Komponenten sind bereit für die Simulation. Alle Beziehungen zwischen den Komponenten werden bestimmt.

Initialisierung des Systems

Vor der Initialisierung des modellierten Systems passt der Benutzer die notwendigen Parameter der Komponenten und die Parameter der Simulation an. Dabei werden die Integrationsschritte der eingebauten Integratoren, der Zeitbereich der Simulation, der Zeitschritt des Simulators, die beobachteten Größen und die variablen Parameter der Netzwerkelemente dargestellt.

Berechnung der Ströme

Bei der Simulation des Modells werden die Zweigströme und die für die Berechnungen notwendigen Spannungen und die Parameter berechnet.

Berechnung der beobachteten Größen

Die für die Beobachtung gewählten Größen werden bei der Anfrage berechnet. Der Benutzer stellt die beobachteten Größen, das Intervall und die Form der Darstellung ein.

Graphische Darstellung der beobachteten Größen

Die beobachteten Größen werden in der passenden Form auf dem Bildschirm dargestellt, gedruckt oder gespeichert (exportiert).

3.4 Modellierung des Netzwerkes mit einem Schalter

3.4.1 Aufgabenstellung

Die Prüfung des Algorithmus für die Berechnung des Netzwerkes mit einem Schalter wird am Beispiel der Modellierung des Kreises in Bild 3.6 dargestellt.

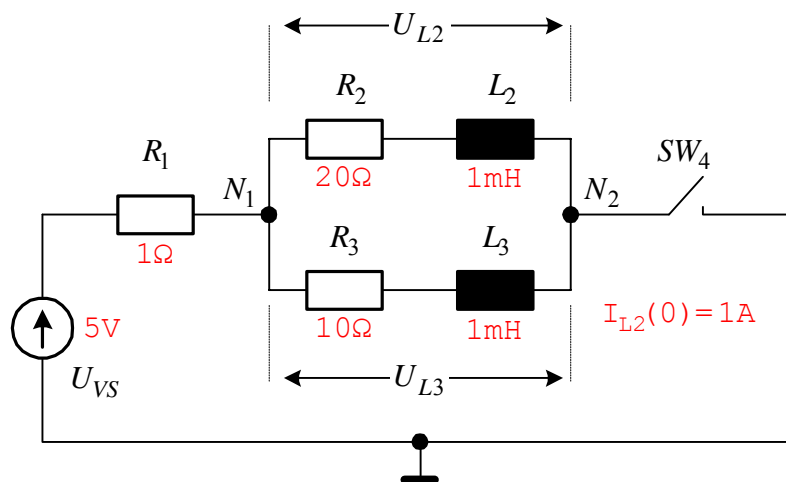


Bild 3.6: Netzwerk mit einem Schalter

Das Netzwerk besteht aus der Spannungsquelle U_{VS} , den Widerständen R_1, R_2, R_3 , den Induktivitäten L_2, L_3 , dem Schalter SW_4 und zwei Knoten N_1, N_2 . Die Spannung der Spannungsquelle hat den Wert:

$$U_{VS} = \begin{cases} 0, & t < 0; \\ 5V, & t > 0. \end{cases} \quad (3.10)$$

Der Schalter zum Zeitpunkt 0 bleibt geschlossen. Zum Zeitpunkt $t_{SW} = 0,5ms$ wird er geöffnet. Der Widerstand des Schalters ist:

$$R_{SW4} = \begin{cases} 0, 1\Omega, & \text{wenn geschlossen;} \\ 1M\Omega, & \text{wenn geöffnet.} \end{cases} \quad (3.11)$$

Die Anfangsströme der Spulen sind: $I_{L2}(0) = 1A$, $I_{L3}(0) = 0$. Alle anderen Parameter der elektrischen Elemente sind im Schaltbild dargestellt.

Die Aufgabe lautet: Berechnung der Ströme und der Spannungen des Kreises im Zeitintervall $t = [0, 1ms]$ mit dem Zeitschritt $\Delta t = 1\mu s$.

3.4.2 Bildung des komponentenorientierten Modells des Netzwerkes

Das komponentenorientierte Modell des Netzwerkes wird auf zwei Strukturblöcke und drei hierarchische Ebenen aufgeteilt (Bild 3.7).

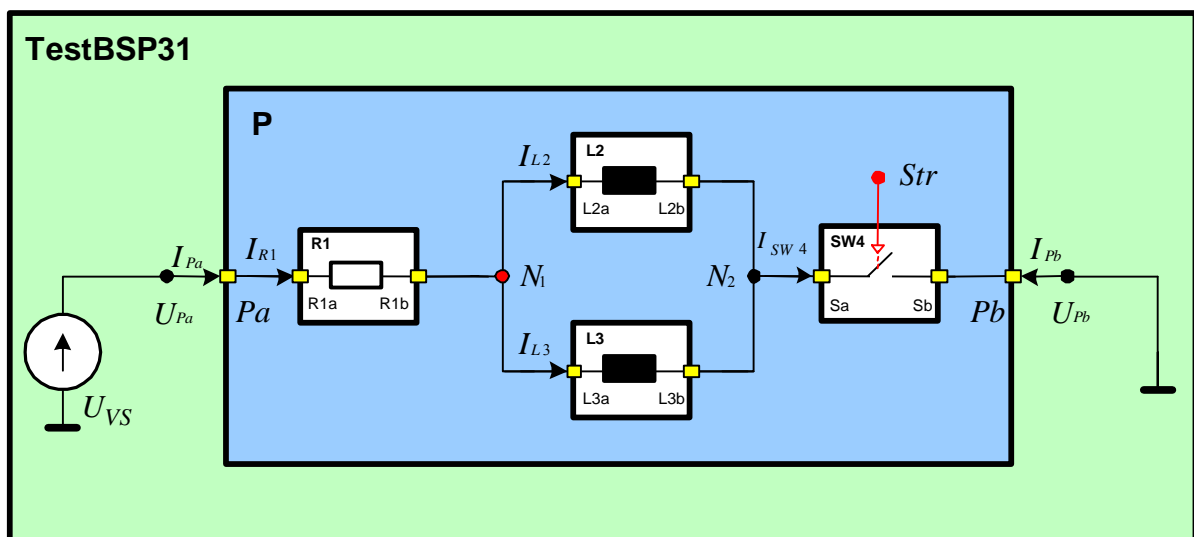


Bild 3.7: Komponentenorientiertes Modell des Netzwerkes

Die Strukturkomponente P mit zwei Pins Pa , Pb enthält die Grundblöcke $R1$, $L2$, $L3$, $SW4$ und Knoten $N1$, $N2$. Die Strukturkomponente $TestBSP31$ enthält die Komponente P und wird mit der Spannungsquelle U_{VS} und der Masse verbunden. Die Grundblöcke bilden die hierarchische Ebene 0, der Block P - die Ebene 1, der Block $TestBSP31$ - die Ebene 2.

Der Referenzpunkt des Blocks P wird im Knoten $N1$ angesetzt. Der Widerstand $R2$ und die Induktivität $L2$ werden mit einem Block „Spule“ $L2$ modelliert. $R3$ und $L3$ werden mit dem Block $L3$ modelliert.

Die Parameter der modellierten Grund- und Strukturblöcke sind in der Tabelle 3.1 dargestellt.

Tabelle 3.1: Parameter der modellierten Grund- und Strukturblöcke

Name des Blocks	Parameter	vordefinierte Werte
$R1$	$U_{R1}^* = 0, R_{R1}^* = R_1, L_{R1}^* = 0$	$R_1 = 1\Omega$
$L2$	$U_{L2}^* = 0, R_{L2}^* = R_2, L_{L2}^* = L_2$	$R_2 = 20\Omega, L_2 = 1\text{mH},$ $I_{L2}(0) = 1\text{A}$
$L3$	$U_{L3}^* = 0, R_{L3}^* = R_3, L_{L3}^* = L_3$	$R_3 = 10\Omega, L_3 = 1\text{mH},$ $I_{L3}(0) = 0$
$SW4$	$U_{SW4}^* = 0, R_{SW4}^* = R_{SW}, L_{SW4}^* = 0$	$R_{SW} = 0,1\Omega$ (Zustand 1), $R_{SW} = 1\text{M}\Omega$ (Zustand 0)
P	$\mathbf{U}_{\mathbf{P}}^* = \begin{pmatrix} U_{Pa}^* \\ U_{Pb}^* \end{pmatrix},$ $\mathbf{R}_{\mathbf{P}}^* = \begin{pmatrix} R_{Paa}^* & R_{Pab}^* \\ R_{Pba}^* & R_{Pbb}^* \end{pmatrix},$ $\mathbf{L}_{\mathbf{P}}^* = \begin{pmatrix} L_{Paa}^* & L_{Pab}^* \\ L_{Pba}^* & L_{Pbb}^* \end{pmatrix}$	-
$TestBSP31$	$\mathbf{U}_{\mathbf{P}} = \begin{pmatrix} U_{Pa} \\ U_{Pb} \end{pmatrix}$	$U_{Pa} = U_{VS},$ $U_{Pb} = 0$

Die Matrizengleichungen für die Strukturkomponente P werden ebenso wie im Beispiel 2.3 formuliert. Die notwendigen Gleichungen nach der symbolischen Lösung in Maple werden im Anhang A zusammengefasst. Die Zustandsgrößen sind die Ströme der Spulen I_{L2}, I_{L3} .

Die Zweigströme und die Knotenspannungen des Blocks P werden durch die Gleichungen (A.6) definiert:

$$\begin{aligned}
 I_{R1} &= \frac{U_{Pa} - U_{N1}}{R_1}, \\
 I_{SW4} &= \frac{U_{N2} - U_{Pb}}{R_{SW}}, \\
 U_{N2} &= U_{Pb} - I_{Pb} R_{SW}, \\
 U_{Pa} &= I_{Pa} R_1 + U_{N1}, \\
 I_{Pb} &= -I_{L2} - I_{L3}.
 \end{aligned} \tag{3.12}$$

Die Gleichungen (3.12) werden in das Modell der Komponente P für die Berechnung der inneren Unbekannten implementiert. Die Grundblöcke „Spulen“ enthalten die Integratoren, die die Ströme der Spulen I_{L2} , I_{L3} integrieren. Die Gleichungen für die Integratoren aus der Lösung der Matrixgleichung (A.1) ergeben sich zu:

$$\begin{aligned} \dot{I}_{L2} &= \frac{U_{N1} - U_{N2} - I_{L2}R_2}{L_2}, \\ \dot{I}_{L3} &= \frac{U_{N1} - U_{N2} - I_{L3}R_3}{L_3}. \end{aligned} \quad (3.13)$$

Die Anfangswerte der Integratoren sind $I_{L2}(0)$, $I_{L3}(0)$. Ein weiterer Integrator gehört zum Block P . Er integriert den Wert von Y , dieser wird dann in den Parameter U^* eingesetzt.

Block TestBSP31

Die Komponente *TestBSP31* der obersten Ebene hat keine innere Komponente außer dem Subblock P mit zwei Pins. Man betrachtet die Stromberechnung auf der obersten Ebene des modellierten Systems für die Komponente P . Die notwendigen Gleichungen werden nach der Bildung und der symbolischen Lösung in Maple für die Komponente *TestBSP31* im Anhang A zusammengefasst.

Die Gleichungen für die Berechnung der Schnittstellenparameter des Blocks P aus (A.11) ergeben sich zu:

$$\begin{aligned} U_{REF} &= U_{Pa} - R_{Paa}^* I_{Pa}, \\ I_{Pa} &= \frac{-U_{Pb} + R_{Pbb}^* I_{Pb} + U_{Pa} + L_{Pbb}^* \dot{I}_{Pb}}{R_{Paa}^*}, \\ \dot{I}_{Pb} &= -\frac{R_{Paa}^* I_{Pb} - U_{Pb} + R_{Pbb}^* I_{Pb} + U_{Pa}}{L_{Pbb}^*}. \end{aligned} \quad (3.14)$$

3.4.3 Simulation

Die Simulation beginnt in einem Integrator des Blocks $L2$ oder $L3$. Für die Integration des Stromes braucht dieser den Wert der Ableitung des Stroms. Die Reihenfolge der Stromberechnung und die Kommunikation zwischen den Komponenten werden in Bild 3.8 dargestellt.

Nach der Berechnung der Ströme kann jede Komponente den Spannungsabfall oder andere innere Größen berechnen, die vom Nutzer für die graphische Darstellung und die Beobachtung gewählt werden.

Das modellierte Netzwerk wurde nach der symbolischen Generierung der notwendigen Gleichungen in den Komponenten des Simulators VSD implementiert. Die Komponenten kommunizieren mittels prototypisch implementierter Schnittstellen.

Für den Vergleich der Ergebnisse der Simulation wird das modellierte Netzwerk in PSPICE simuliert. Bild 3.9 zeigt das Schaltbild in PSPICE.

Die Simulation wird in VSD mit dem Integrationsschritt $\Delta t = 1\mu s$ im Zeitintervall $t = [0, 1ms]$ durchgeführt.

Die in VSD berechneten Ströme und Spannungen sind in Bild 3.10 dargestellt. Die in PSPICE berechneten Ströme und Spannungen sind in Bild 3.11 dargestellt. Der Vergleich des Ergebnisses der Simulation beim Integrationsschritt $\Delta t = 1\mu s$ ergibt den Relativfehler der Berechnung in Höhe von 2%. Das kann über die unterschiedlichen Zeitschritte in VSD und PSPICE erklärt werden. In VSD wird einen konstante Zeitschritt verwendet, PSPICE verwendet einen gesteuerten Zeitschritt.

Der Zeitschritt in PSPICE ändert sich von $1\mu s$ bis $1ns$. Bei der Verkleinerung des Integrationsschrittes in VSD bis zu $\Delta t = 10ns$ wird der Relativfehler kleiner als $0,002\%$. Die Integrationszeit steigt aber sehr schnell (Bild 3.1).

Der Relativfehler hat dieselbe Größe wie bei den Strömen. Unterschiede gibt es nur zum Zeitpunkt der Umschaltung.

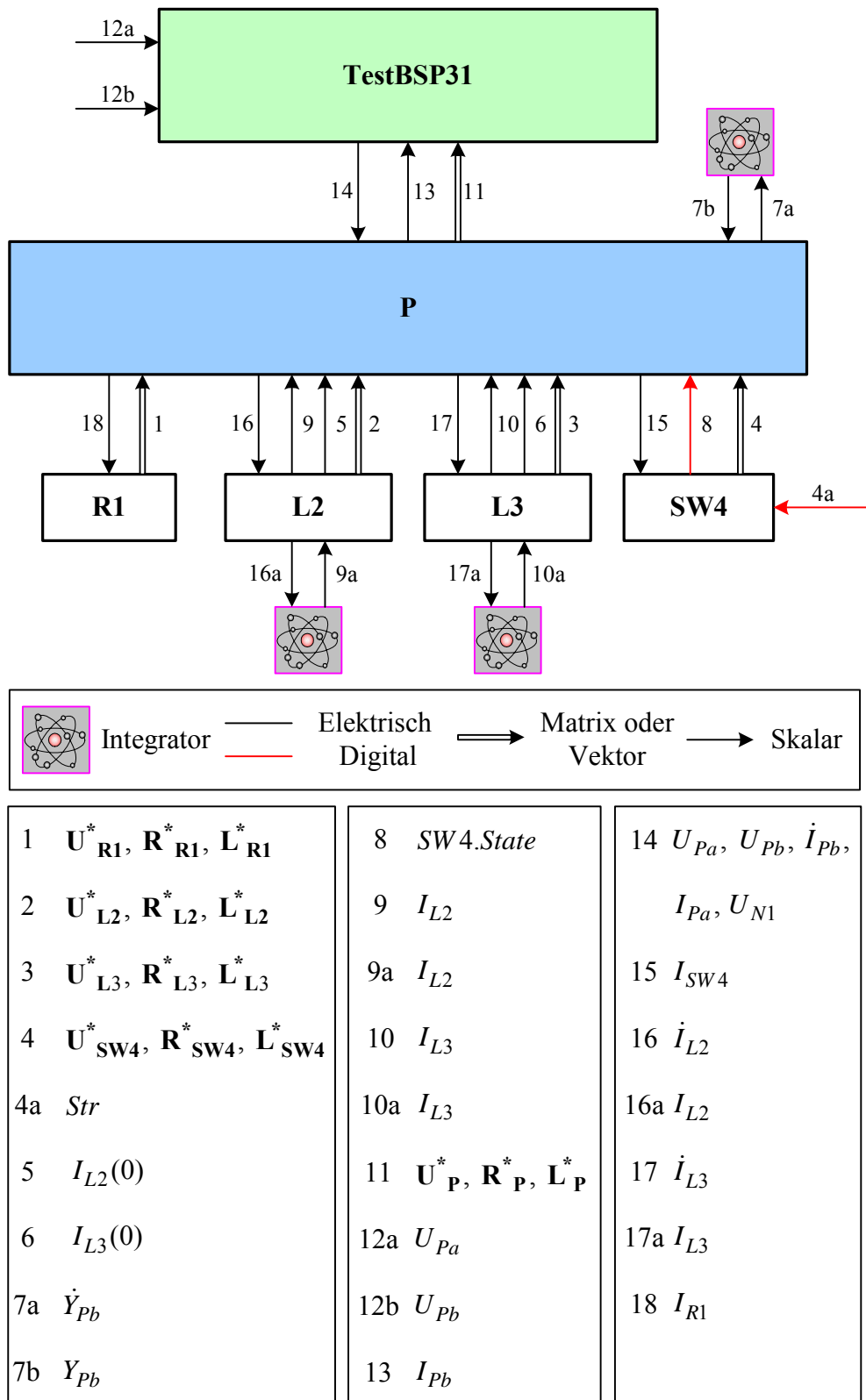


Bild 3.8: Reihenfolge der Stromberechnung

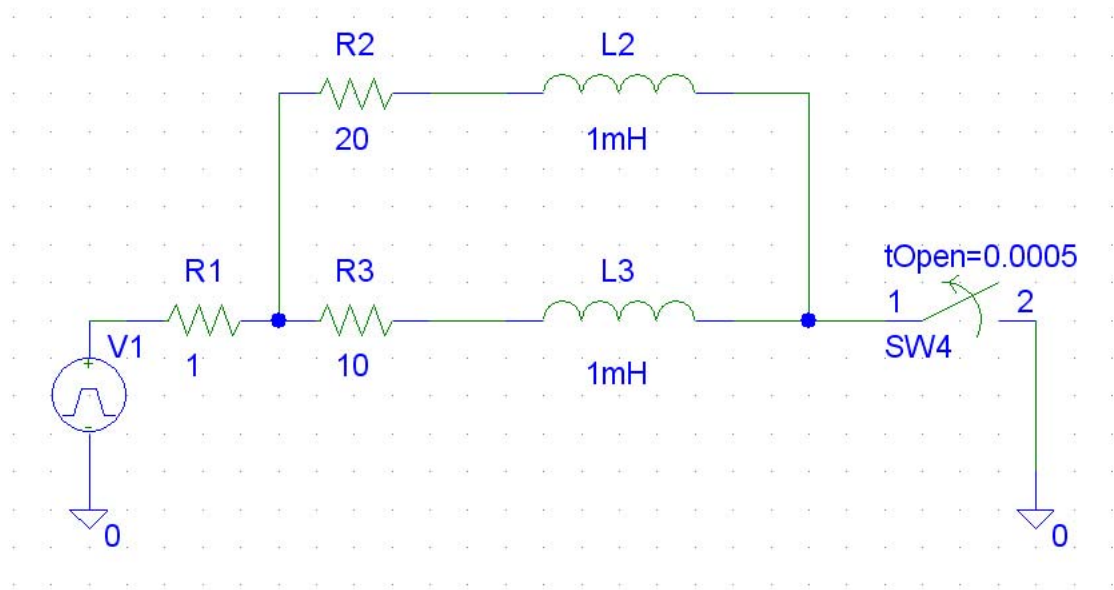


Bild 3.9: Schaltbild des modellierten Netzwerks in PSPICE

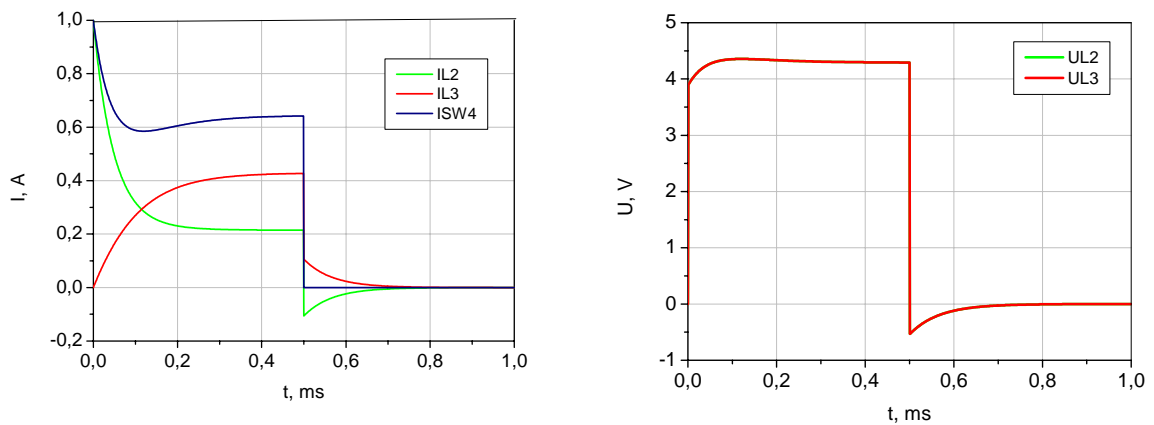


Bild 3.10: Berechnete Zweigströme (VSD)

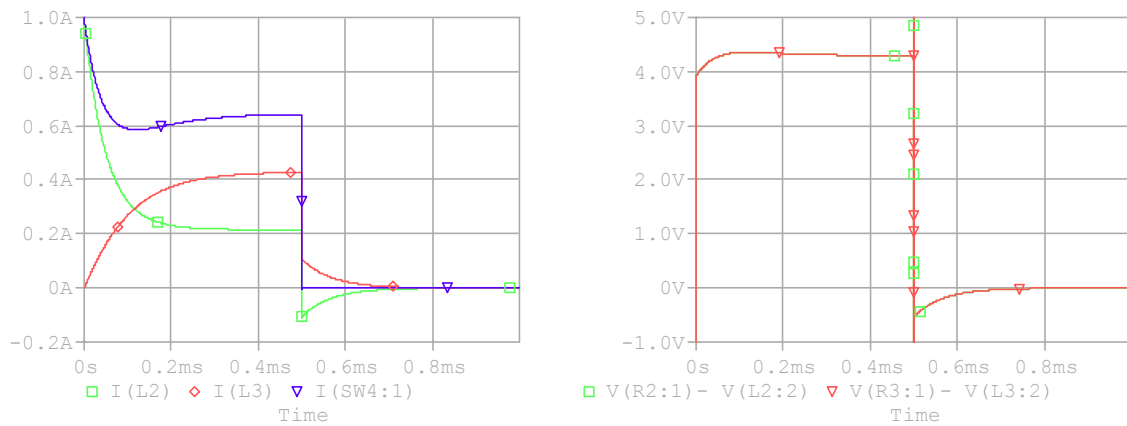


Bild 3.11: Berechnete Zweigströme (PSPICE)

Der ideale Schalter schaltet sofort ohne Verzögerung, deshalb ändert sich die Zeitkonstante des Ausgleichsprozesses sehr schnell. PSPICE berechnet den Vorgang mit einem Gleichungssystem, und im Umschaltzeitpunkt ändert sich die Ableitung des Stroms sehr schnell. Der Spannungsabfall an der Spule ist proportional zur Ableitung des Stroms und ergibt die Spannungsspitzen in Höhe von 2kV (Bild 3.11). Der Ausgleichprozess während der idealisierten Umschaltung wird in PSPICE als gedämpfte Schwingungen berechnet.

Bild 3.12 stellt den Teil des Bildes 3.11 mit einem größeren Zoomfaktor dar.

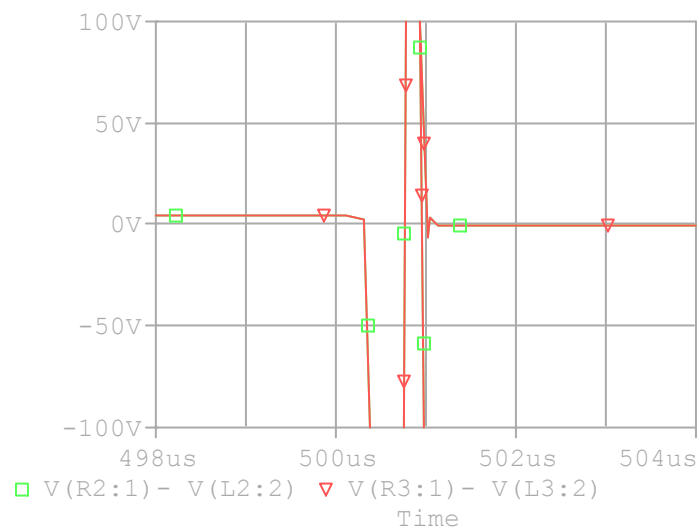


Bild 3.12: Der Ausgleichprozess während der idealisierten Umschaltung (PSPICE)

Die komponentenorientierte Methode des VSD betrachtet beide Fälle separat. Die Impulsstörspitzen während der idealisierten Umschaltung werden vermieden.

4 Eingabe des Netzwerkes

Die Struktur des Netzwerkes und die Parameter der Elemente können aus der Netzliste im SPICE - Format oder aus dem XML - basierten eigenen Format beschrieben werden. Die Netzliste wird eingelesen, analysiert und die elementaren Matrizen werden für das Sparse-Tableau generiert.

Die andere Möglichkeit der Eingabe der Strukturkomponenten ist die visuelle Drag & Drop Technologie in Visio für die Verbindung der Basiskomponenten in die hierarchischen Strukturblöcke. Die Basisblöcke werden aus den Bibliotheken der Standardkomponenten genommen.

4.1 Struktur der SPICE – Netzliste

Ein typisches SPICE – Programm oder Netzliste besteht aus den folgenden Komponenten:

- Titel
- Elementdeklaration
- Kontrolldeklaration
- Ende

Die Elementdeklaration definiert die Elemente im Netzwerk [ATI02], [ALH94]. Die Elementdeklaration besteht aus dem Name des Elements, den Knoten, an den das Element angeschlossen ist und den Werten der Parameter, die das Element elektrisch charakterisieren:

Name Knoten Werte [Parameter]

Der Name des Elements muss mit einem alphabetischen Buchstabe anfangen, der einzigartig für die Elemente und Subblöcke ist. Netzwerkknoten sind die positiven ganzen Zahlen. Der Knoten 0 definiert die Masse des Netzwerkes. Jeder Knoten muss mindestens an zwei Elemente angeschlossen sein. Die Werte des Elements können die nichtnegativen rationalen Zahlen sein. Alle Symbole nach dem Skalierungsfaktor werden ignoriert. Die Buchstaben der

Elemente und die Abkürzungen des Skalierungsfaktors sind in Anhang D dargestellt. Die Parameter, die Defaultwerte gleich Null haben, werden in eckigen Klammern [] geschrieben.

Ein allgemeines Format für die Beschreibung der Widerstände ist

RName N+ N- Wert [TC = TC1,TC2]

wobei der Name mit dem Buchstabe **R** anfangen muss. **N+** und **N-** sind der positive und der negative Knoten des Widerstandes. Der Strom fließt vom positiven Knoten **N+** durch den Widerstand zum negativen Knoten **N-**. **Wert** bestimmt den Wert des Widerstands. Der Wert darf nicht Null sein. **TC1** und **TC2** sind die Temperaturkoeffizienten. Der Defaultwert ist Null.

Ein allgemeines Format für die Beschreibung der linearen Spule ist

LName N+ N- Wert [IC = Anfangsstrom]

wobei der Name mit dem Buchstabe **L** anfangen muss. **N+** und **N-** sind der positive und der negative Knoten der Spule. Der Strom fließt vom positiven Knoten **N+** durch die Spule zum negativen Knoten **N-**. **Wert** bestimmt den Wert der Induktivität. Der Anfangsstrom für den Ausgleichsprozess wird in **IC** festgesetzt.

Ein allgemeines Format für die Beschreibung des linearen Kondensators ist

CName N+ N- Wert [IC = Anfangsspannung]

wobei der Name mit dem Buchstabe **C** anfangen muss. **N+** und **N-** sind der positive und der negative Knoten des Kondensators. Der Strom fließt vom positiven Knoten **N+** durch den Kondensator zum negativen Knoten **N-**. **Wert** bestimmt den Wert der Kapazität. Die Anfangsspannung für den Ausgleichsprozess wird in **IC** festgesetzt.

Ein allgemeines Format für die Beschreibung der unabhängigen Spannungsquelle ist

VName N+ N- [DC Wert] [...]

wobei der Name mit dem Buchstabe **V** anfangen muss. **N+** und **N-** sind der positive und der negative Knoten der Spannungsquelle. Der Strom fließt vom positiven Knoten **N+** durch die Spannungsquelle zum negativen Knoten **N-**. **DC Wert** bestimmt den Wert der Gleichspannung für die Gleichstromanalyse. In [ATI02] werden die anderen Typen der Spannungsquelle definiert.

Ein allgemeines Format für die Beschreibung der unabhängigen Stromquelle ist

IName N+ N- [DC Wert] [...]

wobei der Name mit dem Buchstabe **I** anfangen muss. **N+** und **N-** sind der positive und der negative Knoten der Stromquelle. Der Strom fließt vom positiven Knoten **N+** durch die Stromquelle zum negativen Knoten **N-**. **DC Wert** bestimmt den Wert des Gleichstroms für die Gleichstromanalyse.

Ein allgemeines Format für die Beschreibung der Diode ist

DName N+ N- DMOD

wobei der Name mit dem Buchstabe **D** anfangen muss. **N+** und **N-** sind der positive und der negative Knoten der Diode. Der Strom fließt vom positiven Knoten **N+** durch die Diode zum negativen Knoten **N-**. **DMOD** bestimmt den Typ der Diode. **.MODEL** bestimmt einige Parameter des Modells:

.MODEL DMOD D (...)

Ein allgemeines Format für die Beschreibung des Subkreises (Subblocks) ist

.SUBCKT SUBCIRCUIT_NAME KNOTEN1 KNOTEN2 ...
[PARAMS:NAME=<Wert>]
Subblockdeklaration
.ENDS [SUBCIRCUIT_NAME]

Die Definition des Subblocks fängt mit **.SUBCKT** an. Die Definition des Subblocks endet mit **.ENDS**. Das Symbol für die Subblöcke ist **X**.

Ein allgemeines Format für die Beschreibung des Subblocks ist

XName KNOTEN1 KNOTEN2 ... SUBCIRCUIT_NAME
[PARAMS:NAME=<Wert>]

SUBCIRCUIT_NAME ist der Name des Subblocks, der muss im Netzwerk eingefügt werden. **KNOTEN1 KNOTEN2** sind dieselben Knoten wie in der Subblocksdeklaration **.SUBCKT**.

Bei den in ORCAD PSPICE generierten Netzlisten werden die Namen der Elemente und die Knoten folgend bezeichnet:

- Die Namen des Elements und die Buchstaben der Typen der Elemente werden mit dem Unterzeichen getrennt. Zum Beispiel wird der Widerstand R1 in der Netzliste als R_R1 bezeichnet.
- Die Namen der Knoten werden im Format **\$N_Nummer** bezeichnet.

Bild 4.1 stellt die ORCAD PSPICE Netzliste des Netzwerkes aus Bild 3.12 dar.

```
* Schematics Netlist *
R_R3          $N_0002 $N_0001  10
X_SW4         Uref 0 Sw_tOpen PARAMS: tOpen=0.0005 ttran=0 Rclosed=0.1
+ Ropen=1Meg
R_R2          $N_0002 $N_0003  20
R_R1          $N_0004 $N_0002  1
V_V1          $N_0004  0
+PULSE 0 5 0 0 0 1
L_L3          $N_0001 Uref  1mH IC=0
L_L2          $N_0003 Uref  1mH IC=1
```

Bild 4.1: PSPICE Netzliste des Netzwerkes aus Bild 3.12

Der Kommentar wird mit dem Sternchen ***** angefangen. Ein Schalter wird als Subblock beschrieben.

4.2 Annahmen und Erweiterung der PSPICE – Netzliste für die Zwecke der komponentenorientierten Methode

Für die optimale Nutzung der PSPICE – Netzliste wurden die folgenden Annahmen verwendet. Für jeden Strukturblock muss der Referenzpunkt bestimmt werden. Ein Knoten kann mit dem Namen **NREF** in der Netzliste bezeichnet werden. Der Knoten **NREF** wird als Referenzknoten betrachtet. Falls der Knoten **NREF** nicht in der Netzliste eingetragen wird, dann wird der Knoten mit der maximalen Anzahl angeschlossener Elemente als Referenzknoten betrachtet.

Die Erweiterung der Netzlistedeclaration hilft die Strukturkomponenten für die komponentenorientierte Modellierung zu beschreiben. Die Netzlistedeclaration wird mit speziellen Grundkomponenten erweitert.

Die Grundkomponente „idealer Schalter“ wird mit der folgenden Deklaration bestimmt:

SWName N+ N- Wert1 Wert2

wobei der Name mit der Buchstaben **SW** anfangen muss. Der Strom fließt vom positiven Knoten **N+** durch den Schalter zum negativen Knoten **N-**. **Wert1** bestimmt den Wert des Widerstands des Schalters im geschlossenen Zustand. **Wert2** bestimmt den Wert des Widerstands des Schalters im geöffneten Zustand.

Die Grundkomponente „reaktive Komponente“ wird mit der folgenden Deklaration bestimmt:

ZName N+ N- WertU WertR WertL

wobei der Name mit dem Buchstabe **Z** anfangen muss. Der Strom fließt vom positiven Knoten **N+** durch die Komponente zum negativen Knoten **N-**. Die Werte **WertU**, **WertR**, **WertL** bestimmen die Parameter U^* , R^* , L^* des Blocks.

Die Pins der Komponente werden mit der Deklaration **.EXTERNAL PIN** bestimmt:

.EXTERNAL PIN PINName

wobei **PINName** der Name des Pins ist.

Der Algorithmus der Analyse der ORCAD PSPICE – Netzliste und der Bildung der elementaren Matrizen für das Sparse-Tableau wurde entwickelt und in das MAPLE – Programm implementiert. Die Eingabedatei ist die Textdatei mit der PSPICE - Netzliste. Die Ausgangdatei ist die Textdatei mit den elementaren Matrizen im MAPLE Format.

5 Modellierung des Schaltverstärkers der Piezoansteuerung

Die piezoelektrische Aktor als Stellelement hat in den letzten Jahren ein weit gefächertes Anwendungsfeld gefunden. Seine guten dynamischen Eigenschaften und die hohen Stellkräfte bei kompakter Bauweise machen ihn zu einer attraktiven Alternative für bestehende Lösungen, haben ihm aber auch vielfältige neue Einsatzmöglichkeiten im Bereich der Präzisionsantriebe erschlossen. Bis zu 80% der Aufladeenergie des Aktors wird beim Entladen wieder freigesetzt. Ein wirtschaftlicher Betrieb eines Piezo-Aktors ist deshalb nur mit einem Leistungsverstärker möglich, der die zurückfließende Ladung zwischenspeichern kann und beim nächsten Aufladen wieder nutzt [KAY03].

Am Institut für Mechatronik und Antriebstechnik wurde ein entsprechender Schaltverstärker entwickelt.

5.1 Schaltverstärker in Form einer Ladungspumpe

Für die Ansteuerung von Piezoaktoren wurde ein Schaltverstärker in Form einer Ladungspumpe mit zwei Kreisen entwickelt. Er ist in der Lage, die für das Laden des Aktors benötigte Nennspannung aus einer Spannungsquelle der niedrigen Ebene zu erzeugen und die beim Entladen zurückfließende Ladung zu speichern [KAS98-4], [KAS99-3], [IMA99], [GNA05].

Die Ladungspumpe enthält zwei Kreise mit Spulen unterschiedlicher Induktivität. Durch die unterschiedlichen Induktivitäten entspricht der Schaltverstärker hohen Anforderungen sowohl an die Dynamik wie auch an die Genauigkeit der Verstellung der Aktorspannung. Eine Darstellung der vollständigen Topologie des leistungselektronischen Teils des Schaltverstärkers zeigt Bild 5.1.

Die Ladungspumpe umfasst die beiden Funktionseinheiten Aufladekomponente UP und Entladekomponente DN [KAY03]. Beide Teile sind als Schaltregler ausgeführt. Sie bestehen aus je einer Spule, einem Schaltelement und einer Diode.

Zum Aufladen des Aktors wird der Schalter SI (MOSFET- Transistor o.ä.) für kurze Zeit geschlossen, bis der sich in der Spule LI aufbauende Strom einen gewünschten Wert erreicht hat. Dann wird der Schalter geöffnet und die in der Spule gespeicherte Energie fließt nun über

die Diode $D1$ in den Piezoaktor PZ . Dieser Vorgang wird wiederholt, bis die Aktorspannung ihren Sollwert angenommen hat.

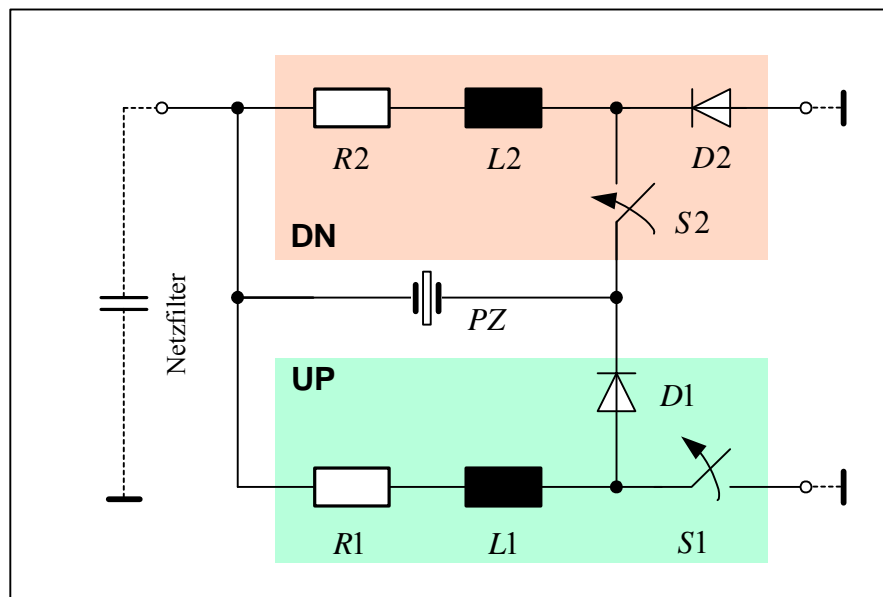


Bild 5.1: Ladungspumpe

Zum Entladen des Aktors wird der Schalter $S2$ geschlossen. Der Aktor entlädt sich über die Spule $L2$. Wenn der Spulenstrom einen gewünschten Wert erreicht hat, wird der Schalter wieder geöffnet. Die in der Spule gespeicherte Energie fließt nun über die Diode $D2$ auf die Eingangsklemmen und lädt den Kondensator im Netzfilter (Energiespeicherung). Auch dieser Vorgang wird solange wiederholt, bis der Sollwert der Aktorspannung erreicht wird.

In beiden Fällen entsteht nach dem Schließen des Transistors zunächst ein Spulenstrom, der sich anschließend über die Diode abbaut. Statt diskreten Dioden kann die Freilaufdiode des anderen Transistors verwendet werden [GNA05].

Um den Aktor definiert umzuladen, müssen die Schaltimpulse für die beiden Transistoren innerhalb einer Reaktionszeit von weniger als einer Mikrosekunde berechnet werden. Das wird durch Implementierung des Regelalgorithmus auf einem FPGA erreicht [IMA99], [GNA05].

Die Auswahl der Bauelemente wird in [GNA05] ausführlich betrachtet. Mit den MOSFET-Transistoren stehen entsprechende Schalter mit einem Widerstand von $R_{ON} = 0.15\Omega$ zur Verfügung. Der Entwurf des Schaltverstärkers erfolgte für den Betrieb von Aktoren mit einer

Kapazität von $1\mu\text{F}$ bis $10\mu\text{F}$. Die Spule $L1$ hat eine Induktivität von $550\mu\text{H}$, die Spule $L2$ hat eine Induktivität von $140\mu\text{H}$.

5.2 Bildung der Komponenten der Ladungspumpe

Die Ladungspumpe wird mit dem Konzept der komponentenorientierten Modellierung betrachtet. Die Schaltung in Bild 5.1 kann in drei funktionale hierarchische Ebenen aufgeteilt werden. Die Grundblöcke bilden die Ebene 0. Die Strukturkomponenten UP und DN bilden die Ebene 1. Die Strukturkomponente $DRIVER$ enthält die Blöcke UP , DN . Die Grundkomponente PZ und bildet die Ebene 2.

5.2.1 Bildung der Komponente UP

Der Block UP besteht aus dem Widerstand $R1$, der Spule $L1$, dem Schalter $S1$, der Diode $D1$, sowie den Knoten N_1 und N_{REF} (Bild 5.2).

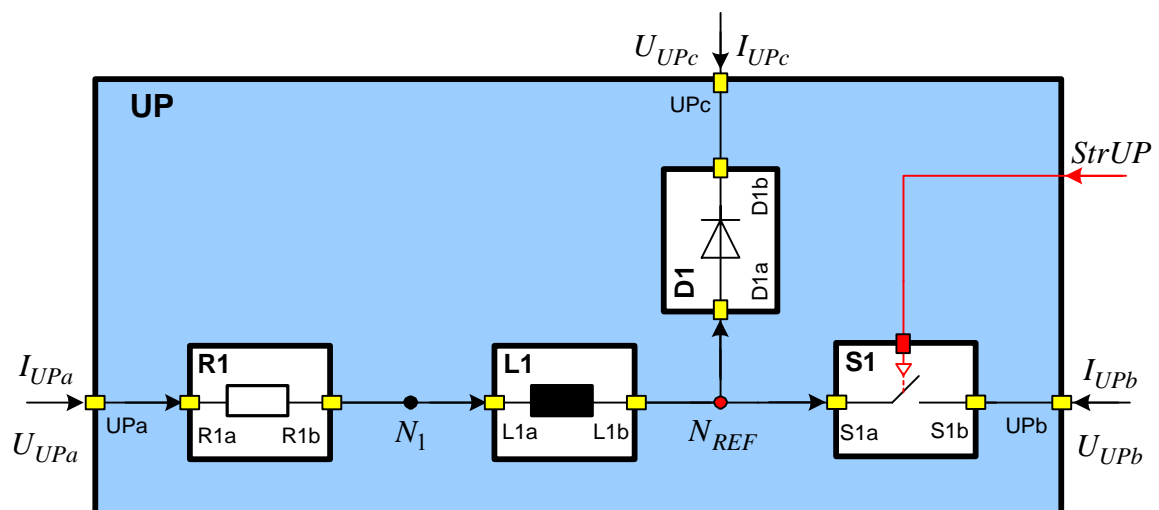


Bild 5.2: Komponentenorientiertes Modell des Blocks UP

Die Strukturkomponente UP hat drei Pins UPa , UPb , UPc . Der Referenzpunkt ist im Knoten N_{REF} eingesetzt. Die Grundkomponenten $R1$, $L1$, $D1$, $S1$ bilden die hierarchische Ebene 0, der Block UP bildet die Ebene 1.

Die Modelle der Diode $D1$ und des Schalters $S1$ haben zwei innere Zustände: entweder „gesperrt“ oder „leitend“. Die Umschaltung des Schalters $S1$ wird mit dem Signal $StrUP$ gesteuert.

Die Parameter der Grundblöcke der Komponente UP sind in Tabelle 5.1 dargestellt.

Tabelle 5.1: Die Parameter der Grundblöcke der Komponente UP

Name des Blocks	Parameter	Werte
$R1$	$U_{R1}^* = 0, R_{R1}^* = R_1, L_{R1}^* = 0$	$R_1 = 0,1\Omega$
$L1$	$U_{L1}^* = 0, R_{L1}^* = 0, L_{L1}^* = L_1$	$L_1 = 140\mu\text{H}, I_{L1}(0) = 0$
$D1$	$U_{D1}^* = V_D, R_{D1}^* = R_D, L_{D1}^* = 0$	$R_D = 3\Omega, V_D = 0,3\text{V}$
$S1$	$U_{S1}^* = 0, R_{S1}^* = R_{SW}, L_{S1}^* = 0$	$R_{SW} = 0,15\Omega$ (leitet), $R_{SW} = 10\text{M}\Omega$ (gesperrt)

Für das Testmodell der Diode wird die Kennlinie der Diode D1N3600 verwendet.

Die Netzliste der Komponente UP für die Ermittlung der Parameter U^* , R^* , L^* und der Komponentengleichungen wird in Bild 5.3 dargestellt.

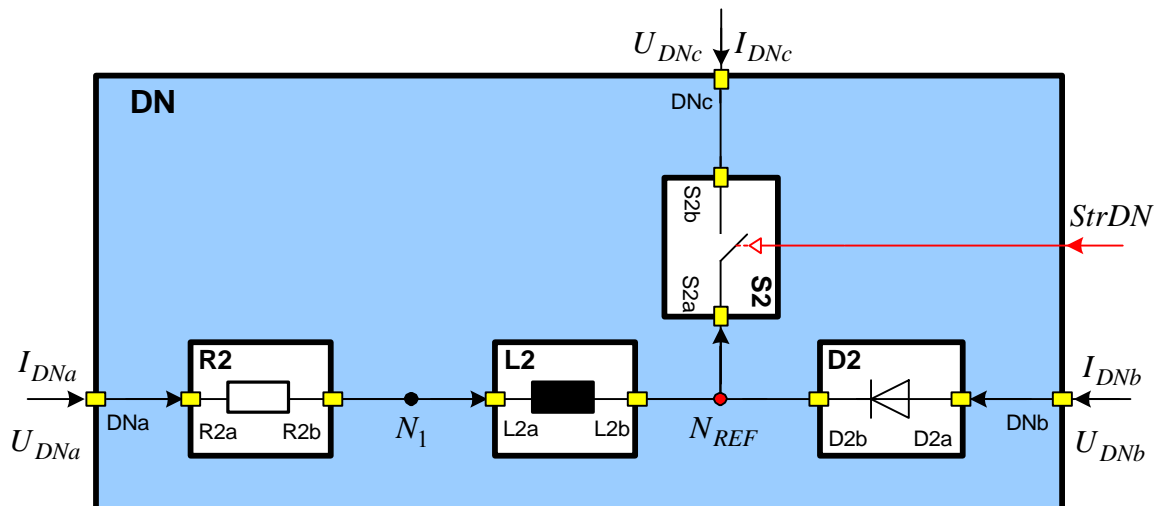
```
* Block UP *

.EXTERNAL INPUT UPa
.EXTERNAL INPUT UPb
.EXTERNAL INPUT UPc
R_R1 UPa N_1 0.1
L_L1 N_1 N_REF 0.00014
SW_S1 N_REF UPb 0.15 10Meg
D_D1 N_REF UPc Dt
```

Bild 5.3: Netzliste des Blocks UP

5.2.2 Bildung der Komponente DN

Der Block DN besteht aus dem Widerstand $R2$, der Spule $L2$, dem Schalter $S2$, der Diode $D2$, sowie den Knoten N_1 und N_{REF} (Bild 5.4).

Bild 5.4: Komponentenorientiertes Modell des Blocks *DN*

Die Strukturkomponente *DN* hat drei Pins *DNa*, *DNb*, *DNc*. Der Referenzpunkt ist im Knoten N_{REF} eingesetzt. Die Grundkomponenten *R2*, *L2*, *D2*, *S2* bilden die hierarchische Ebene 0, der Block *DN* - die Ebene 1.

Die Modelle der Diode *D2* und des Schalters *S2* haben zwei innere Zustände: entweder „gesperrt“ oder „leitend“. Die Umschaltung des Schalters *S2* wird mit dem Signal *StrDN* gesteuert.

Die Parameter der Blöcke der Komponente *DN* sind in Tabelle 5.2 dargestellt.

Tabelle 5.2: Parameter des Blocks *DN*

Name des Blocks	Parameter	Werte
<i>R2</i>	$U_{R2}^* = 0, R_{R2}^* = R_2, L_{R2}^* = 0$	$R_2 = 0,1\Omega$
<i>L2</i>	$U_{L2}^* = 0, R_{L2}^* = 0, L_{L2}^* = L_2$	$L_2 = 550\mu\text{H}, I_{L2}(0) = 0$
<i>D2</i>	$U_{D2}^* = V_D, R_{D2}^* = R_D, L_{D2}^* = 0$	$R_D = 3\Omega, V_D = 0,3\text{V}$
<i>S2</i>	$U_{S2}^* = 0, R_{S2}^* = R_{SW}, L_{S2}^* = 0$	$R_{SW} = 0,15\Omega$ (leitend), $R_{SW} = 10\text{M}\Omega$ (gesperrt)

Die Netzliste der Komponente *DN* für die Ermittlung der Parameter U^* , R^* , L^* und der Komponentengleichungen wird in Bild 5.5 dargestellt.

```

* Block DN *

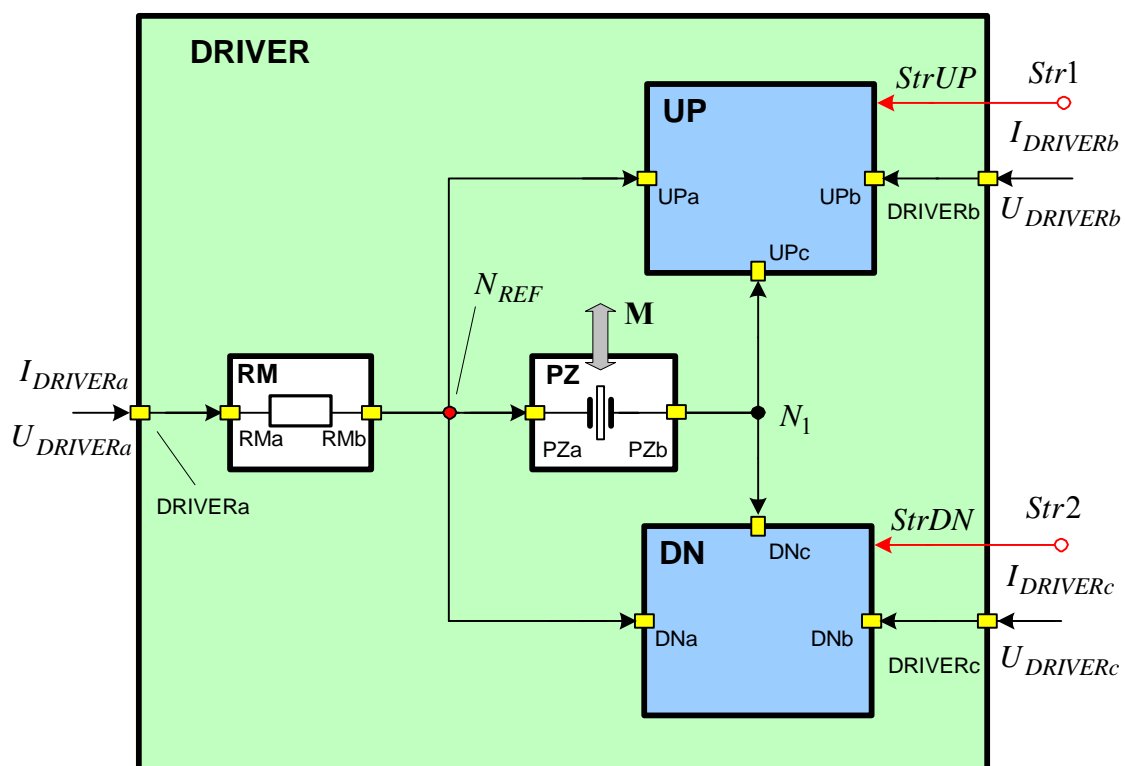
.EXTERNAL INPUT DNa
.EXTERNAL INPUT DNb
.EXTERNAL INPUT DNC
R_R2 DNa N_1 0.1
L_L2 N_1 N_REF 0.00055
SW_S2 N_REF DNC 0.15 10Meg
D_D2 DNb N_REF Dt

```

Bild 5.5: Netzliste des Blocks *DN*

5.2.3 Bildung der Komponente *DRIVER*

Der Block *DRIVER* besteht aus dem Piezoaktor *PZ*, dem Widerstand *RM*, den Strukturblöcken *UP* und *DN* sowie den Knoten N_1 und N_{REF} (Bild 5.6).

Bild 5.6: Komponentenorientiertes Modell des Blocks *DRIVER*

Die Grundkomponenten *PZ*, R_M und die Strukturkomponenten *UP* und *DN* bilden die hierarchische Ebene 1, der Block *DRIVER* bildet die Ebene 2.

Die Netzliste der Komponente DN für die Ermittlung der Parameter U^* , R^* , L^* und der Komponentengleichungen wird in Bild 5.7 dargestellt.

```
* Block DRIVER *  
  
.EXTERNAL INPUT DRIVERa  
.EXTERNAL INPUT DRIVERb  
.EXTERNAL INPUT DRIVERc  
R_RM DRIVERa N_REF 0.1  
PZ_PZ N_REF N_1 1u 0.1  
X_UP N_REF DRIVERb N_1  
X_DN N_REF DRIVERc N_1
```

Bild 5.7: Netzliste des Blocks $DRIVER$

Der Anschluss $DRIVERa$ muss an eine Spannungsquelle angeschlossen werden. Die Anschlüsse $DRIVERb$ und $DRIVERc$ müssen auf Masse gelegt werden.

5.2.4 Simulation

Für die bessere Anschaulichkeit wird der Betrieb der Ladungspumpe in zwei separaten Phasen betrachtet. Die Ergebnisse der Berechnung einer Phase werden als Anfangsbedingungen für die andere Phase verwendet. In der ersten, der Aufladungsphase, werden die Komponenten UP und PZ im Aufladungszweig simuliert. In der zweiten, der Entladungsphase, werden die Komponenten DN und PZ im Entladungszweig simuliert.

Der Piezoaktor ist das gemeinsame Element für die beiden Zweige. Die Komponenten UP und PZ wirken über die im Piezoaktor gespeicherte Energie zusammen. Die Berechnung beider Phasen können in der beliebigen Reihenfolge wiederholt werden.

Das modellierte Netzwerk wurde nach der symbolischen Generierung der notwendigen Gleichungen in den Komponenten des Simulators VSD implementiert. Die Komponenten kommunizieren mittels einer prototypisch implementierten Schnittstelle.

Aufladungsphase

Für den Vergleich der Ergebnisse der Simulation wird das modellierte Netzwerk in PSPICE simuliert. Bild 5.8 zeigt das Schaltbild des Aufladungszweigs im PSPICE. Der Piezoaktor *PZ* wird als die Schaltung aus dem Widerstand *RPZ* und dem Kondensator *CPZ* modelliert.

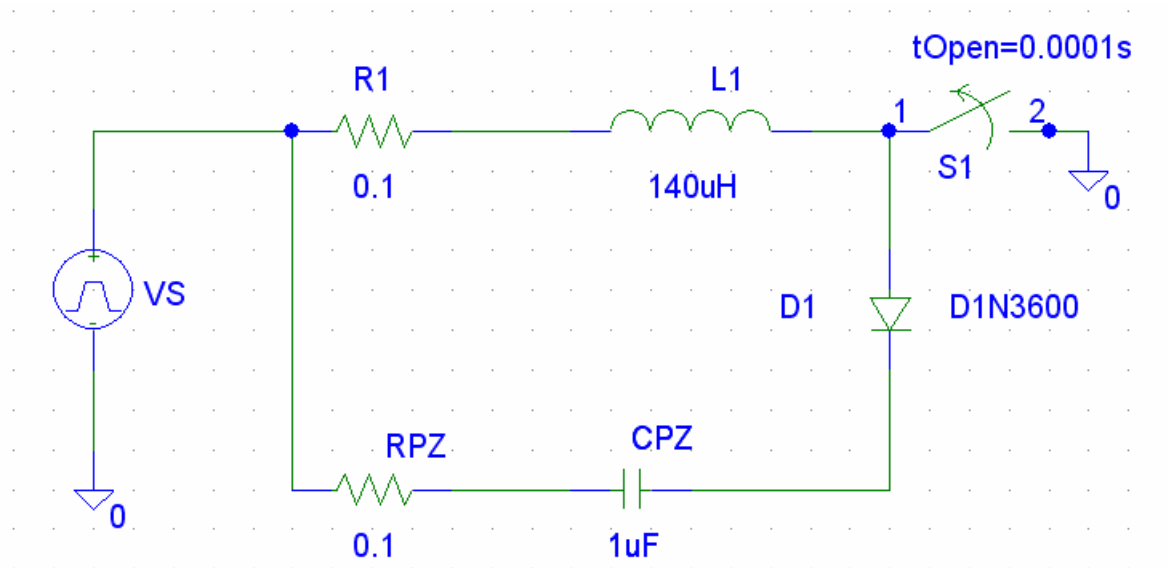


Bild 5.8: Schaltbild des Aufladungszweigs in PSPICE

Die Spannungsquelle *VS* liefert eine Gleichspannung von 5V. Der Schalter *S1* bleibt bis zum Zeitpunkt 0,1ms geschlossen und die Spule *L1* wird aufgeladen. *D1* ist gesperrt. Danach wird *S1* geöffnet, der Spulenstrom fließt durch die geöffnete Diode *D1* und lädt den Kondensator *CPZ* auf.

Die Simulation wird in VSD mit dem Integrationsschritt $\Delta t = 1\mu s$ im Zeitintervall $t = [0, 0,5\text{ms}]$ durchgeführt. PSPICE verwendet den adaptiven Integrationsschritt.

Die in VSD berechneten Ströme und Spannungen sind in Bild 5.9 dargestellt. Bild 5.10 zeigt die in PSPICE berechnete Ströme und Spannungen.

Der Kondensator wird bis zur Spannung $U_{AUF} = 33,63\text{V}$ aufgeladen. Die Aufladungskurven sind identisch, aber die Spannung an der Spule bei der Berechnung in VSD enthält keine Störschwingungen im Umschaltunkt. Dies wird mit der optimalen Behandlung des Ausgleichsprozesses in VSD erklärt. Bei der Sperrung der Diode oder des Schalters steigt der Widerstand um den Faktor 10^6 und die Zeitkonstante des Ausgleichsprozesses der Umschaltung sinkt dem entsprechend. In PSPICE wird die Zeitschritt für die korrekte

Stromintegration automatisch verkleinert bis auf 0,01..1ns und bei der Berechnung der idealen Umschaltung entstehen die Störschwingungen. In VSD wird der schnelle Ausgleichsprozess der Umschaltung vermieden (s. Kapitel 3) und alles mit dem Integrationsschritt 1 μ s wesentlich ökonomischer berechnet.

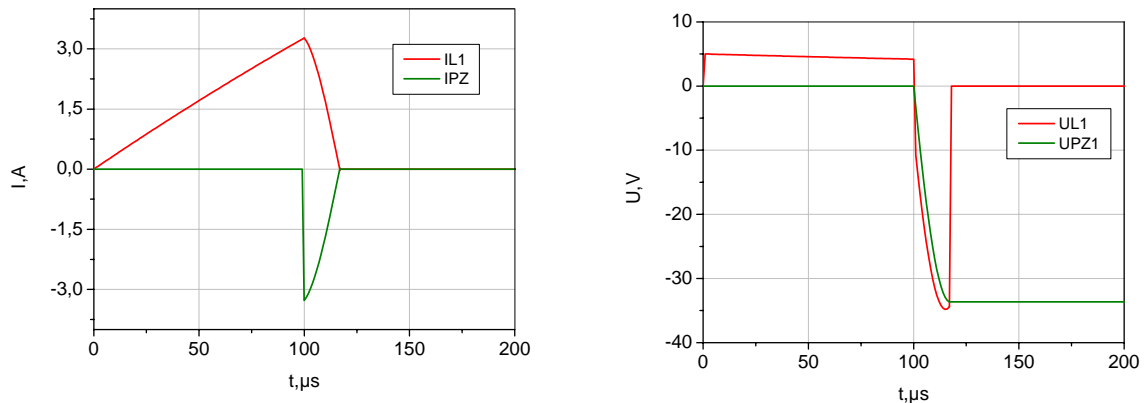


Bild 5.9: Berechnete Ströme und Spannungen in VSD (Aufladungsphase)

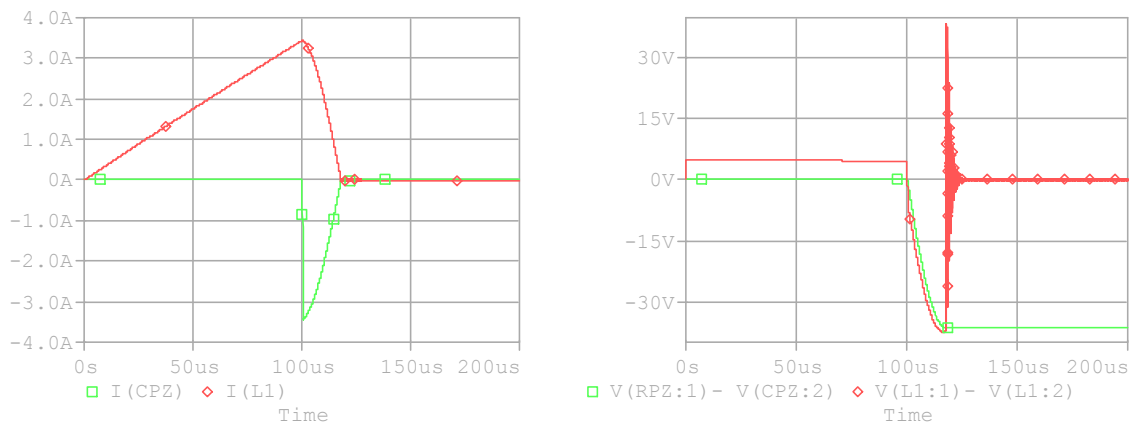


Bild 5.10: Berechnete Ströme und Spannungen in PSPICE (Aufladungsphase)

Entladungsphase

Bild 5.11 zeigt das Schaltbild des Entladungszweigs in PSPICE.

Die Spannungsquelle V_S liefert eine Gleichspannung von 5V. Der Schalter S_2 bleibt bis zum Zeitpunkt 0,2ms geschlossen. D_2 ist gesperrt. Der Kondensator CPZ ist bis $U_{AUF} = 33,63V$ aufgeladen. Danach wird S_2 geöffnet, CPZ entlädt sich auf die Spule L_2 . Danach fließt der Induktivitätsstrom durch die geöffnete Diode D_2 .

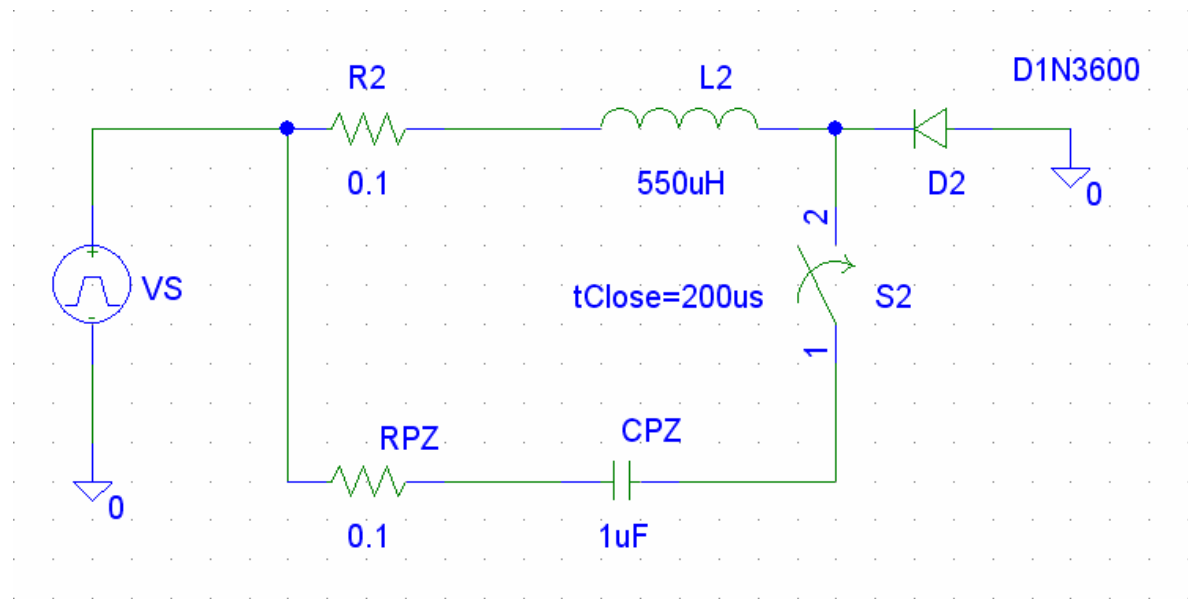


Bild 5.11: Schaltbild des Aufladungszeigs in PSPICE

Die Simulation wird in VSD mit dem Integrationsschritt $\Delta t = 1\mu s$ im Zeitintervall $t = [0, 0,5ms]$ durchgeführt. PSPICE verwendet den adaptiven Integrationsschritt.

Die in VSD berechneten Ströme und Spannungen sind in Bild 5.12 dargestellt. Bild 5.13 zeigt die in PSPICE berechnete Ströme und Spannungen.

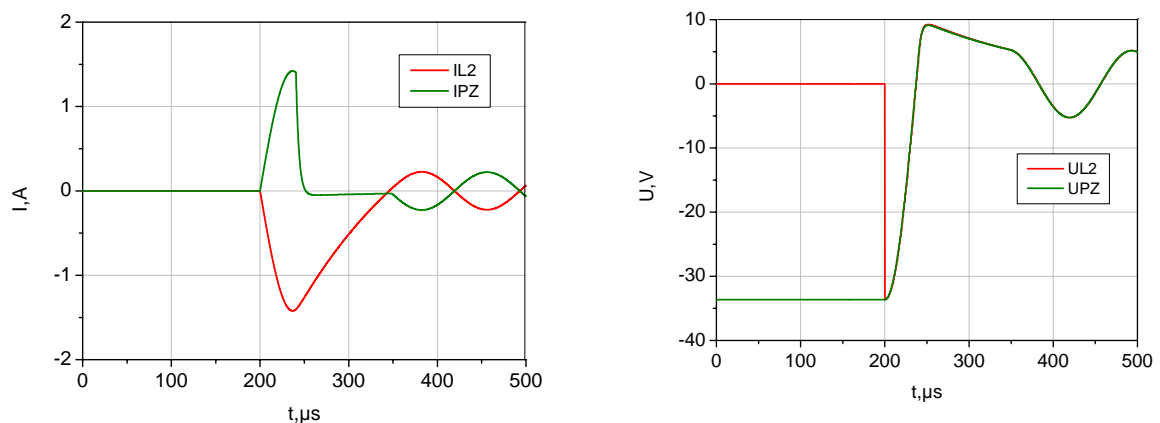


Bild 5.12: Berechnete Ströme und Spannungen in VSD (Entladungsphase)

Die Entladungskurven sind identisch, aber die Spannung an der Spule bei der Berechnung in VSD enthält keine Störschwingungen im Umschaltzeitpunkt. Für die Berechnung der Entladungsphase können dieselben Bemerkungen wie für die Aufladungsphase gemacht werden.

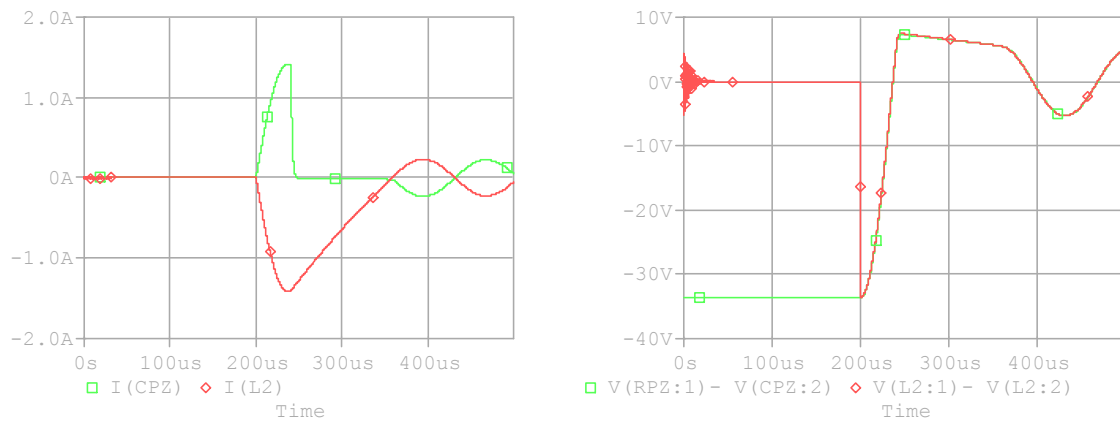


Bild 5.13: Berechnete Ströme und Spannungen in PSPICE (Entladungsphase)

Die Modellierung des Schaltverstärkers der Piezoansteuerung mittels der in dieser Arbeit beschriebenen Methode und der Vergleich mit der Simulation in PSPICE zeigt, dass die vorgeschlagene komponentenorientierte Methode für die typischen mechatronischen Schaltungen sehr gut anwendbar ist.

6 Zusammenfassung

6.1 Ergebnisse

Modellierung und Simulation haben mit der Einführung der Informationstechnik in den letzten fünfzig Jahren einen Großteil der Ingenieurwissenschaften durchdrungen. Simulationsmodelle werden an Stelle realer Systeme herangezogen, um die inneren Zusammenhänge eines Systems zu begreifen und Prognosen über dessen Verhalten zu machen. Im Vergleich zu realen Experimenten erfordern virtuelle Experimente häufig einen erheblich geringeren finanziellen und zeitlichen Aufwand, da die Modellierung virtueller Prototypen in der Regel deutlich weniger aufwändig ist als der Aufbau realer Prototypen.

Das Schlüsselthema der Mechatronik stellt nach wie vor die Integration verschiedener Technologien und Wissensgebiete dar. Durch diese interdisziplinäre Aufgabenstellung, ergeben sich Schwierigkeiten bei der Entwicklung mechatronischer Produkte, da die Modellierung nicht durchgängig erfolgen kann. Bis heute existieren hierzu keine kommerziell brauchbare Tools, die mehr als einen Teilbereich oder die Kopplung zweier Teilbereiche gut abdecken.

In der Arbeit wurden die entwickelten theoretische Grundlagen und spezielle Punkte der komponentenorientierten Modellierung elektrischer Systeme als ein Teil eines mechatronischen Systems dargestellt. Die Prinzipien und Strategien zur Umsetzung einer komponentenorientierten Modellierung technischer Systeme wurde betrachtet.

Die entwickelte Methode verwendet das Modulkonzept. Die Module werden als Software-Module oder -Klassen dargestellt und kommunizieren miteinander mittels eindeutig definierter Schnittstellen. Die Schnittstellen sind so definiert, dass die Komponenten aus verschiedenen Domänen zusammenwirken können. Aus informationstechnischer Sicht werden die Module in hierarchischen Strukturen gebildet. Die einzelnen Module können leicht ersetzt oder ausgetauscht werden. Die inneren Größen der Module bleiben während der Simulation in den Modulen gekapselt.

Die Komponenten können mit zwei Methoden erzeugt werden: entweder aus einer SPICE ähnlichen Netzliste (oder XML- basierter Netzliste) oder mit Komponentenabbildung in Visio. Der Import der SPICE ähnlichen Netzliste in MAPLE- anwendbare Form wurde erzeugt.

Der Algorithmus der Analyse der ORCAD PSPICE – Netzliste und der Bildung der elementaren Matrizen und Vektoren entsprechend der vorgeschlagenen Methodik wurde entwickelt und in der MAPLE – Programmiersprache implementiert. Die Eingabedatei ist die Textdatei mit der PSPICE - Netzliste. Die Ausgangsdatei ist die Textdatei mit den elementaren Matrizen im MAPLE Format. Die SPICE – Netzliste Struktur wurde erweitert und die bestimmten Annahmen wurde durchgeführt für die bessere Anwendbarkeit bei der komponentenorientierten Modellierung der elektrischen Systeme.

Die Algorithmen für die symbolische Bildung der Matrizen und Vektoren, die symbolische Lösung der Matrixgleichungen und die Ermittlung der Komponentenparameter wurden in MAPLE – Programmiersprache implementiert. Die symbolisch berechneten Gleichungen und die Parameter der Modelle werden bei der Bildung der Module verwendet.

Die Implementierung der Methode wurde in Visual C und Visual Basic auf objektorientierte Weise durchgeführt. Die prototypische Schnittstelle, Verbindungen und Funktionen sind manuell entwickelt und implementiert worden. Die prototypische Bibliothek der Grundkomponenten wurde zusammengesetzt. Die Bibliothek enthält einen Widerstand, eine Spule, einen Kondensator, eine Spannungsquelle, einen Schalter, eine Diode und einen Piezoaktor. Eine Strukturkomponente kann aus den Grundkomponenten zusammengesetzt und mittels Simulationssoftware VSD gebildet werden. Die Strukturkomponenten können mit VSD simuliert werden.

Die Behandlung der Schaltvorgänge wurde untersucht. Der Einfluss des Integrationsschritts auf die Berechnungszeit wurde untersucht und Methoden der Behandlung sehr kleiner Integrationsschritte wurden betrachtet. Ein Konzept für die Behandlung der schnell laufenden Ausgleichsprozesse wurde vorgeschlagen. Die Systeme mit den Umschaltungen werden mit den vorgeschlagenen Annahmen bei ausreichender Genauigkeit ökonomischer als in PSPICE berechnet.

Neben dem theoretischen Teil, der Prinzipien und Herangehensweisen zum Inhalt hat, wurden auch die bei der praktischen Umsetzung und bei der Implementierung gemachten Erfahrungen vorgestellt. Die Methode wurde mit einer großen Anzahl verschiedener Beispielschaltungen getestet, wobei ihre gute Anwendbarkeit nachgewiesen wurde.

Das vereinfachte Modell des Verstärkers der Ansteuerschaltung eines Piezoaktors als typisches Beispiel einer elektrischen Komponente eines mechatronischen Systems wurde aufgebaut und simuliert. Das Modell ist auch für die Vergleichzwecke in PSPICE simuliert und mit Testergebnissen aus weiterer wissenschaftlicher Arbeit am IMAT verglichen worden. Die Simulation liefert in beiden Fälle vergleichbare Ergebnisse, obwohl das

Komponentenmodell wesentlich einfacher aufgebaut ist und entsprechend weniger Rechenaufwand benötigt. Die Modellierung des Schaltverstärkers der Piezoansteuerung mittels der in dieser Arbeit beschriebenen Methode und der Vergleich mit der Simulation in PSPICE zeigt, dass die vorgeschlagene komponentenorientierte Methode für typische mechatronische Schaltungen sehr gut anwendbar ist.

Folgende wesentlichen Vorteile können zusammenfassend genannt werden:

- Schnelle Erstellung von Systemmodellen, leichte Bedienbarkeit, gute Aufbereitung der Ergebnisse.
- Intuitives Verständnis von Modellen und Modulen.
- Parallele Erstellung von Modellkomponenten durch unterschiedliche Nutzer.
- Einfache Erweiterbarkeit der Modelle.
- Leichte Ersetzbarkeit und Umtauschbarkeit von Komponenten.

6.2 Ausblick

In weiteren Schritten soll die prototypisch entwickelte Schnittstelle und die prototypische Bibliothek der Komponenten ausgebaut werden. Die implementierten Algorithmen müssen in Visual Studio Net Code implementiert werden.

Die Prinzipien der Verteilung nach schnellen und langsamen Gruppen der Module sollen untersucht werden sowie die Fragen der Parallelisierung der Berechnungen.

Die optimale Behandlung nichtlinearer Netzwerke mit stromabhängigen Widerständen durch iterative numerische Verfahren ist ein notwendiger Schritt, so dass nur bei Vorliegen eines stromabhängigen Widerstands in einem Modul (lokal oder über die Schnittstelle importiert) eine Iteration/Strukturumschaltung ausgeführt werden muss. Parametrierungsdaten sowie alle für die Simulation notwendigen Daten müssen über Interfaces der Komponenten zur Verfügung gestellt werden.

Die Verbindung der mechanischen, elektrischen und elektromechanischen Module, in ein gemeinsames System, muss weiterentwickelt und implementiert werden.

Die Entwicklung einer allgemeinen Methodik zur Abbildung und Repräsentation dynamischer elektrischer Modellkomponenten in 3D-CAD-Modellen, unter Nutzung des dort vorhandenen

Objektvorrats, muss ausgeführt werden. Funktionsmodelle könne z.B. in Form von Netzlisten aus CAE-Modellen importiert werden.

Die Entwicklung einer allgemeinen Methodik muss für die numerische Simulation von gekoppelten Komponentenmodellen ausgeführt werden. Die Komponentenstruktur der Modelle bleibt während der Simulation erhalten, so dass kritische Effekte wie z.B. Haftreibungs- oder Schaltvorgänge in einzelnen Komponenten gekapselt werden können und so auch Integrationsschrittweiten und –verfahren komponentenspezifisch festgelegt werden können.

Literatur

- ADM Adamski, Dirk: Komponentenorientierte Simulation mechatronischer Systeme. VDI, Reihe 10, Nr682. ISBN 3-18-368280-9.
- ALH94 Al-Hashimi, Bashir: The art of simulation using PSpice: analog and digital / Bashir Al-Hashimi. p. cm. – CRC Press, Inc.: Boca Raton, 1994.
- ALI99 Alimeling, J.H. Hammer, W.P.: PLECS-piece-wise linear electrical circuit simulation for Simulink Power Electron. & Electrometrol. Lab., Swiss Fed. Inst. of Technol., Zurich; This paper appears in: Power Electronics and Drive Systems. PEDS '99, 1999.
- ATI02 Attia, John Okyere: PSPICE and MATLAB for electronics: an integrated approach / John Okyere Attia. P. cm. – Boca Raton; London; New York; Washington D. C., 2002.
- BEC97 Becker, U.; Schwarz, P.: Modellierung und Simulation nichtelektrischer Systeme. Elektronik 1/1997, 70-75 und 2/1997, 82-86, 1997.
- BER T. Bertram, F. Bekes, R. Greul, O. Hanke, C. Haß, J. Hilgert, M. Hiller, O. Öttgen, P. Opgen-Rhein, M. Torlo, D. Ward: Modelling and simulation for mechatronic design in automotive systems. Control Engineering Practice, Vol. 11, No. 2, pp. 177-188.
- BER01 T. Bertram, P. Opgen-Rhein: Modellbildung und Simulation mechatronischer Systeme - Virtueller Fahrversuch als Schlüsseltechnologie der Zukunft. Automotive Electronics. // ATZ/MTZ Automotive Engineering Partners-Sonderausgabe 11, 20-25, 2001.
- BUR04 Burgmeier, S.; Handschuch, K.; Ramthum, C.; Sprothen, V.: Mit einem Mausclick. // Wirtschaftswoche, Heft 39, S. 20-30, 2004.
- CAS CASPOC: <http://www.caspoc.com/showpage.php3>
- CLA03 Clauß, Chr.; Haase, J.; Schneider, P.; Schwarz, P. : Einsatz von standardisierten Verhaltensbeschreibungssprachen für die Modellierung in der Mikrosystemtechnik. // 9.GMM-Workshop, Bielefeld, 31.03.2003 - 01.04.2003, 25-32, 2003.
- DAM04 V. Damic und J. Montgomery: Mechatronics by bond graphs: an object-oriented approach to modelling and simulation. Springer-Verlag Berlin Heidelberg New York, 2004. ISBN 3-540-42375-3.

- EDM90 Edminister, Joseph A.: Elektrische Netzwerke/ Joseph A. Edminister, Übers. u. dt. Bearb.: Henriette Hartl. – 2. überarb. u. erw. Aufl. – Hamburg; New York [u. a.]: McGraw-Hill, 1990.
- ENG03 Enge, O.: Modelling of Electromechanical Systems With Variable Structure Using the Linear Complementarity Problem. // 11th International Symposium on Applied Electromagnetics and Mechanics - ISEM'03, Versailles, France, May 12-14, 2003, Proceedings, pp. 208-209, 2003.
- ENG04 Enge, O.: Modelling of Electromechanical Systems With Variable Structure Using the Linear Complementarity Problem. // Int. J. of Applied Electromagnetics and Mechanics, 19(2004)1-4:25-29, IOS Press, 2004.
- ENM Enge, O.; Maißer, P.: Modelling Electromechanical Systems with Electrical Switching Components Using the Linear Complementarity Problem, to appear in: Journal Multibody System Dynamics, 26 pages.
- FOL94 Föllinger, Otto; Dörrscheidt, Frank; Klittsch, Manfred: Regelungstechnik: Einführung in die Methoden und ihre Anwendung/ Otto Föllinger. Unter Mitw. von Frank Dörrscheidt und Manfred Klittsch. – 8., überarb. Aufl. – Heidelberg: Hüthig, 1994.
- GAU Gausemeier, Jürgen; Riepe, Bernd; Lückel, Joachim: Integrativer Maschinenbau - auf dem Weg zu den Produkten von morgen. (Hrsg.: Krause, F.-L.; Tang, T.; Ahle, U.: Innovationsforum - Virtuelle Produktentstehung ; Berlin (S. S. 315-335).
- GNA05 Gnad, Gunnar: Ansteuerkonzept für piezoelektrische Aktoren/
Internet: <http://diglib.uni-magdeburg.de/Dissertationen/2005/gungnad.pdf>
- HAH98 Hahn, Martin; Lückel, Joachim; Naumann, Rolf; Rasche, Rainer: Ein Objektmodell für den Mechatronikentwurf. ASIM '98, 12. Symposium Simulationstechnik; Zürich, 1998.
- HAN00 Hahn, M.; Koch, T.: CAMeL – Ein Werkzeug zum integrierten CAD-gestützten Entwurf mechatronischer Systeme. SIM2000, Simulation im Maschinenbau, Dresden, 2000.
- HEI98 Heimann, Bodo: Mechatronik: eine Einführung in die Komponenten zur Synthese zbd Methoden zur Analyse mechatronischer Systeme/ von Bodo Heimann; Wilfried Gerth; Karl Popp. [Unter Mitarb. Von Ulrich Schmucher]. – München; Wien: Fachbuchverl. Leipzig im Carl-Hanser-Verl., 1998.

- HEL02 Hellgren, J.: Modelling of Hybrid Electric Vehicles in Modelica for Virtual Prototyping. // Proceedings, 2nd International Modelica Conference, March, 18.-19. 2002, Oberpfaffenhofen, pp 247-256, 2002.
- JUN Junker, Frank, Lückel, Joachim: A Systematics of Modelling Mechatronic Systems. // 1st MATHMOD/IMACS Symposium on Mathematical Modelling, Wien.
- IMA99 Neue Piezoansteuerung mit FPGA, Online-Dokumentation des IMAT der Universität Magdeburg. Internet-Link: <http://imat.mb.uni-magdeburg.de/piezo/> (Stand 10/99)
- JUS94 Justus, Otto: Berechnung linearer und nichtlinearer Netzwerke mit PSpice-Beispielen/ Otto Justus. – Leipzig; Köln: Fachbuchverl., 1994.
- KAS96 Kasper, R.: Objectoriented Modelling and Simulation of Mechatronic Systems. // Symposium on interaction between dynamics and control in advanced mechanical systems, Eindhoven University of Technolgy, Tagungsband, S. 167 - 176, Eindhoven, 1996.
- KAS96-2 Kasper, R.; A. Wagner, J. Heinrich: Modelling und Simulation of Piezoelectric Actuators with Way Enlargement // 5th International Conference on New Actuators ACTUATOR 96, Bremen, 1996.
- KAS97 Kasper, R.; Koch, W.; Sczyrba, Ch.: Object-Oriented Software Techniques in Simulation of Mechatronic Systems: Towards Distributed Hardware-in-the-Loop-Simulation using Standard Hard- and Software. // 15th IMACS World Congress on Scientific Computation, Modelling and Applied Mathematics, Tagungsband V, S. 137 - 142, Berlin, 1997.
- KAS98 Kasper, R.; Koch, W.; Wolf, A.: Event-Based Simulation of Mixed Continuous and Discrete Components of Mechatronic Systems. // 31st ISATA "Simulation, Virtual Reality and Supercomputing; Automotive Applications", Proceeding of the Conference, S. 31 – 38, 1998.
- KAS98-1 Kasper, R.; W. Heinemann; A. Wagner: Modelling and Control for piezoelectric Actuators for High Speed Applications. // MOVIC, ETH Zürich, 1998.
- KAS99 Kasper, R.; Wagner, A.: Modellgestütztes Reglerkonzept für piezoelektrische Antriebe. // 4. Magdeburger Maschinenbautage, 22.-23.9.1999, Magdeburg, 1999.
- KAS03 Kasper, R.; Y. Yefimenko: Komponentenorientierte Modellbildung elektrischer Systeme am Beispiel der Ansteuerschaltung eines Piezoaktors. //

6. *Magdeburger Maschienebau-Tage: Tagungsband*, Otto-von-Guericke-Universität Magdeburg. [Hrsg.: Roland Kasper ...].– Berlin: Logos-Verl., 2003.
- KAS04 Kasper, Roland; Yefimenko, Yevgen: Component-oriented simulation of electrical systems in the field of mechatronic. // Proceedings of the 5th EUROSIM Congress on Modelling and Simulation, ESIEE Paris, Marne la Valee, France, 2004.
- KAS05 Kasper, R.; Yefimenko, Y.: Modelling and Simulation of Electrical Parts of Mechatronic Systems // 7. *Magdeburger Maschienebau-Tage: Tagungsband*, Otto-von-Guericke-Universität Magdeburg. [Hrsg.: Roland Kasper ...].– Berlin: Logos-Verl., 2005.
- KAY03 Kayser, Andreas: Komponentenorientierte Modellierung elektrischer und mechatronischer Systeme. Fortschr.-Ber. VDI Reihe 20 Nr. 368. Düsseldorf: VDI Verlag 2003.
- KIE94 Kielkowski: Ron M. Inside SPICE: overcoming the obstacles of circuit simulation / Ron M. Kielkowski, McGraw-Hill, Inc, - New York, 1994.
- LUN87 K. Lunze: Berechnung elektrischer Stromkreise: Arbeitsbuch; 132 Aufgaben mit Lösungen/ Verl. Technik, Berlin 1987. 308S.
- MAI03 Maisser, P.: Mathematical Modelling of Mechatronic Systems. // 4th Polish-German Mechatronic Workshop, 30.06. - 01.07., Suhl 2003.
- MAS93 Massobrio Giuseppe, Antognetti Paolo: Semiconductor device modelling with SPICE/ Giuseppe Massobrio, Paolo Antognetti – 2nd. Ed. p. cm., McGraw-Hill, Inc, 1993.
- MAT98 Mattsson, S.E., Elmqvist, H., Otter, M.: Physical system modeling with Modelica. // Control Engineering Practice, vol. 6, pp. 501-510, 1998.
- MAT98-2 Mattsson, S.E., Elmqvist, H., Broenink, J.F.: Modelica: An International effort to design the next generation modelling language. // Journal A, Benelux Quarterly Journal on Automatic Control, vol. 38:3, 16-19, September 1997. Special issue on Computer Aided Control System Design, CACSD, 1998.
- OTT03 Otter, M.; Elmqvist, H. and Mattson S.E.: The New Modelica MultiBody Library // Proceedings, 3rd International Modelica Conference, November, 3.-4. 2003, Linköping, Schweden, pp 247-256.

- PEL01 Pelz, Georg: Modellierung und Simulation mechatronischer Systeme: vom Chip- und Systementwurf mit Hardwarebeschreibungssprachen/ Georg Pelz. – Heiderberg: Hüthig, 2001.
- PLE PLECS - Piece-wise Linear Electrical Circuit Simulation for Simulink: <http://www.plexim.com/downloads/index.html>
- PSI PSIM: <http://www.powersimtech.com/>
- PSP PSPICE 9:
<http://www.cadencepcb.com/products/downloads/PSpicestudent/default.asp>
- SIM SIMPLORER: <http://www.ansoft.com/products/em/simplorer/>
- TIL00 Tiller M.; Bowles P.; Elmqvist H.; Brück D.; Mattson S.E.; Möller A.; Olsson H. and Otter M.: Detailed Vehicle Powertrain Modeling in Modelica. Proceedings, Modelica Workshop 2000, October, 23.-24., Lund, Schweden.
- UNB90 R. Unbehauen: Elektrische Netzwerke; eine Einführung in die Analyse/ 3., Neubearbeitete und erweiterte Auflage, Berichtigter Nachdruck, - Berlin, 1990.
- YEF03 Yefimenko. Y.; Kasper, R.: Komponentenorientierte Modellbildung mechatronischer Systeme. // Машиностроение и техносфера XXI века. Сборник трудов международной научно-технической конференции в г. Севастополе 8-14 сентября 2003г.Т4. С. 103-106, – Донецк, 2003.
- YEF05 Yevgen Yefimenko; Roland Kasper: Generalized Calculation Of Interface Parameters In the Component-oriented Method of Modelling Of Electrical Systems. // Машиностроение и техносфера XXI века. Сборник трудов международной научно-технической конференции в г. Севастополе 12-17 сентября 2005г. – Донецк, 2005.

Anhang A Symbolische Lösung der Gleichungen des Netzwerks mit einem Schalter

Blocks P – Schalter $SW4$ ist geschlossen

Sparse-Tableau Matrizengleichung für den Block P ergibt sich:

$$\begin{aligned}
 & \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -1 & 0 \\ 0 & -1 \\ 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} I_{Pa} \\ I_{Pb} \end{pmatrix} + \begin{pmatrix} U_{N1} \\ -U_{N1} \\ -U_{N1} \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \\
 & + \begin{pmatrix} R_1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & R_2 + L_2 s & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & R_3 + L_3 s & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & R_{SW} & -1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & -1 & -1 & 1 & 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} I_{R1} \\ I_{L2} \\ I_{L3} \\ I_{SW4} \\ U_{N2} \\ U_{Pa} \\ U_{Pb} \end{pmatrix} = \begin{pmatrix} 0 \\ L_2 \cdot I_{L2}(0) \\ L_3 \cdot I_{L3}(0) \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \tag{A.1}
 \end{aligned}$$

Das System der Gleichungen für den Block P aus (A.1) ergibt sich:

$$\begin{aligned}
 & R_1 \cdot I_{R1} + U_{N1} - U_{Pa} = 0, \\
 & (R_2 + L_2 s) \cdot I_{L2} - U_{N1} + U_{N2} = L_2 \cdot I_{L2}(0), \\
 & (R_3 + L_3 s) \cdot I_{L3} - U_{N1} + U_{N2} = L_3 \cdot I_{L3}(0), \\
 & R_{SW} \cdot I_{SW4} - U_{N2} + U_{Pb} = 0, \\
 & I_{R1} - I_{Pa} = 0, \\
 & -I_{SW4} - I_{Pb} = 0, \\
 & -I_{L2} - I_{L3} + I_{SW4} = 0.
 \end{aligned} \tag{A.2}$$

Die Anwendung *GaussCalc* und *ParamExtr* auf (A.1) ergibt die Parameter U^* , R^* , L^* des Blocks P . Die Symbolischen Ausdrücke der Parameter sind zu lang und sie werden hier nicht dargestellt.

Die Differentialgleichungen der Spulen ergeben sich:

$$\begin{aligned} \dot{I}_{L2} &= \frac{U_{N1} - U_{N2} - I_{L2}R_2}{L_2}, \\ \dot{I}_{L3} &= \frac{U_{N1} - U_{N2} - I_{L3}R_3}{L_3}. \end{aligned} \quad (\text{A.3})$$

Die Anfangswerten der Integratoren sind: $I_{L2}(0)$, $I_{L3}(0)$.

Die Matrizengleichung (A.1) nach dem Algorithmus *VarCalc* ergibt sich:

$$\begin{aligned} &\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & -1 \end{pmatrix} \cdot \begin{pmatrix} I_{Pa} \\ U_{Pb} \\ I_{L2} \\ I_{L3} \end{pmatrix} + \begin{pmatrix} U_{N1} \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \\ &+ \begin{pmatrix} R_1 & 0 & 0 & -1 & 0 \\ 0 & R_{SW} & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} I_{R1} \\ I_{SW4} \\ U_{N2} \\ U_{Pa} \\ I_{Pb} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \end{aligned} \quad (\text{A.4})$$

Das System der Gleichungen aus (A.4) ergibt sich:

$$\begin{aligned} R_1 \cdot I_{R1} + U_{N1} - U_{Pa} &= 0, \\ R_{SW} \cdot I_{SW4} - U_{N2} + U_{Pb} &= 0, \\ I_{R1} - I_{Pa} &= 0, \\ I_{SW4} + I_{Pb} &= 0, \\ -I_{L2} - I_{L3} + I_{SW4} &= 0. \end{aligned} \quad (\text{A.5})$$

Die Lösung der Matrizengleichung (A.4) Nach dem Algorithmus *GaussCalc* ergibt sich

$$\begin{aligned}
I_{Pb} &= -I_{L2} - I_{L3}, \\
U_{Pa} &= I_{Pa}R_1 + U_{N1}, \\
U_{N2} &= U_{Pb} - I_{Pb}R_{SW}, \\
I_{SW4} &= \frac{U_{N2} - U_{Pb}}{R_{SW}}, \\
I_{R1} &= \frac{U_{Pa} - U_{N1}}{R_1}.
\end{aligned} \tag{A.6}$$

Stromberechnung im Block *TestBSP31*

Die Matrizengleichung für den Block *TestBSP31*:

$$\begin{pmatrix} R_{Paa}^* & 0 \\ 0 & R_{Pbb}^* \\ 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} I_{Pa} \\ I_{Pb} \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & L_{bb}^* \\ 0 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} U_{REF} \\ \dot{I}_{Pa} \\ \dot{I}_{Pb} \end{pmatrix} + \begin{pmatrix} U_{Pa}^* \\ U_{Pb}^* \\ 0 \end{pmatrix} = \begin{pmatrix} U_{Pa} \\ U_{Pb} \\ 0 \end{pmatrix}. \tag{A.7}$$

Das System der Gleichungen für den Block *TestBSP31* aus (A.7) ergibt sich:

$$\begin{aligned}
R_{Paa}^* I_{Pa} + U_{REF} &= U_{Pa}, \\
R_{Pbb}^* I_{Pb} + L_{Pbb}^* \dot{I}_{Pb} + U_{REF} &= U_{Pb}, \\
\dot{I}_{Pa} + \dot{I}_{Pb} &= 0.
\end{aligned} \tag{A.8}$$

Die modifizierte nach der *TopMod* - Funktion Matrizengleichung ergibt sich:

$$\begin{pmatrix} 0 \\ R_{Pbb}^* \\ 1 \end{pmatrix} \cdot (I_{Pb}) + \begin{pmatrix} 1 & R_{Paa}^* & 0 \\ 1 & 0 & L_{bb}^* \\ 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} U_{REF} \\ I_{Pa} \\ \dot{I}_{Pb} \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} U_{Pa} \\ U_{Pb} \\ 0 \end{pmatrix}. \tag{A.9}$$

Das System der Gleichungen für den Block *TestBSP31* aus (A.9) ergibt sich:

$$\begin{aligned}
R^*_{Paa}I_{Pa} + U_{REF} &= U_{Pa}, \\
R^*_{Pbb}I_{Pb} + \dot{L}^*_{Pbb}\dot{I}_{Pb} + U_{REF} &= U_{Pb}, \\
I_{Pa} + I_{Pb} &= 0.
\end{aligned} \tag{A.10}$$

Nach der Anwendung der Funktion **GaussCalc** auf (A.9) ergibt sich:

$$\begin{aligned}
U_{REF} &= U_{Pa} - R^*_{Paa}I_{Pa}, \\
I_{Pa} &= \frac{-U_{Pb} + R^*_{Pbb}I_{Pb} + U_{Pa} + \dot{L}^*_{Pbb}\dot{I}_{Pb}}{R^*_{Paa}}, \\
\dot{I}_{Pb} &= -\frac{R^*_{Paa}I_{Pb} - U_{Pb} + R^*_{Pbb}I_{Pb} + U_{Pa}}{\dot{L}^*_{Pbb}}.
\end{aligned} \tag{A.11}$$

Anhang B: Elementdeklaration in SPICE

Tabelle B.1: Name des Elements und das entsprechenden Element in der SPICE Netzliste

B	GaAs FET Transistor
C	Kapazität
D	Diode
E	Spannungsgesteuerte Spannungsquelle
F	Stromgesteuerte Stromquelle
G	Spannungsgesteuerte Stromquelle
H	Stromgesteuerte Spannungsquelle
I	Unabhängige Stromquelle
J	FET Transistor
K	Transformator
L	Induktivität
M	MOSFET Transistor
Q	Bipolar Transistor
R	Widerstand
S	Spannungsgesteuerter Schalter
T	Übertragungsleitung
V	Unabhängige Spannungsquelle
X	Subnetzwerk (Subblock)

Tabelle B.2: Auszüge aus den Abkürzungen des SPICE - Skalierungsfaktors

Suffix	Metrik Präfix	Multiplizierungsfaktor
Meg	Mega	10^6
K	Kilo	10^3
M	Milli	10^{-3}
U	Micro	10^{-6}
N	Nano	10^{-9}